STAT

Page Denied

Next 1 Page(s) In Document Denied

Czechoslovakian Academy of Sciences

Laboratory for Mathematical Machines

MACHINES FOR PROCESSING INFORMATION

PART I

MATHEMATICAL MACHINES

[redacted]

STAT

Publication of the Czechoslovakian Academy of Sciences

Prague 1953

[redacted]

STAT

The Manual was Compiled by

A.Svoboda

K.Bem, V.Cerny, H.Jermar, Z.Korvas, K.Kristoufek, J.Marek, J.Oblonsky, O.Pokorna, Z.Pokorny, J.Raichl, F.Svoboda, H.Snelerova, M.Sterbova, M.Valach, V.Vysin

[redacted]

STAT

INTRODUCTION

In this volume, lectures to the workers of the Laboratory of Mathematical Machines of the Czechoslovakian Academy of Sciences held at the Working Conference in Dome by science worker J.E.Purkyne in December 1952 are published. The next volume will contain works in the field of computers for processing data submitted by the above-mentioned Laboratory. The acceptance of the works is the province of the Science Council of the Laboratory for Mathematical Machines.

The main purpose of the first volume is to acquaint a wide public with an outline of the work carried on at the Laboratory for Mathematical Machines in the field of automatic computing. The text of the manual is the work of the staff of the Laboratory. The results of the original works were reviewed, and the most important items selected for publication.

The formulation of the central automatic calculation, its codes, the method of construction of the computation are the work of Antonin Svoboda.

The adaptation of the codes in conformity with the constructive requirements of the machines and the extension of the codes necessary for time-saving are the work of Vaclav Cerny.

The examples of the manual were prepared by:

Jan Oblonsky (The computing of cos x), Olga Pokorna (The Geometry of Problems in Optics), Zdenek Pokorny (The Computing Problems in Optics), Jiri Raichl (The Computing of Solutions of Differential Equations).

The formulation of the multiplication perforations and their codes is the work of Antonin Svoboda.

The investigation and execution of the multiplication perforations are the work of the Aritma Research Institute.

The methods of solving the problems described in the second part of the manual and the applied symbols (Operating Tables) are the work of Jindrich Mark.

The staff of the Laboratory for Mathematical Machines had to overcome many basic difficulties. Thanks are due to all who contributed their active support, particularly the Czechoslovakian Academy of Sciences, in whose framework the Laboratory for Mathematical Machines has been successfully developed, and the Mathematical Institute of the Czechoslovakian Academy of Sciences, especially its Director, Academician E.Cechovi, who has lent his unselfish support to this development for a number of years. Thanks are also due to the research workers of Aritma for willing and devoted collaboration.

Thanks are also due to the staff of the Laboratory for Mathematical Machines, and particularly to those members whose collaboration facilitated the preparation of this Manual. Prof. Dr. Hruska carried out the critical scientific examination of the manuscript. Special thanks are due to the Prague printers, shop 05, their Manager K.Wickovi and especially their type setters, who composed even the most difficult parts of the text and tables unusually carefully and in the shortest time.

Prague, 31 December 1952

Collective LMS

STAT

INTRODUCTION TO METHOD OF OPERATION AT AUTOMATIC CALCULATION

CHAPTER I

AUTOMATIC CALCULATION

1.1. Survey

Calculation, logarithmic and other tables are the oldest aids to rapid numerical calculation. In later years computers were designed and developed for completely automatic calculation.

The employment of computers is today so widespread that everybody has at least a clear idea of their use.

A computer has a keyboard for the digits and various operation keys for the solution of operations with digits. When operating the calculating machine, the operator reads from the formula the numbers on which he has to operate, sets them in the machine, carries out the necessary operation, reads off the results, and inserts it into the formula. The transfer of the numbers from the formula to the machine and of the result from the machine to the formula is not as difficult and tedious a job as the execution of arithmetical operations with pencil on paper. The computer carrying out such arithmetical operations rapidly is desirable not only because it accelerates the work but also because it facilitates and improves it by excluding calculating errors.

An automatic computer is a machine which carries out automatically a large number of arithmetical operations, automatically manipulates the numbers on which it operates, and automatically adjusts itself to the operating procedure.

It is incorrect to describe the computer without the last of the mentioned characteristics. According to this incorrect description, the automatic computer is based on the simple mechanization of the working procedure of the calculator using the calculating machine. Such an automatic calculation would mean only the acceleration of numerical calculation carried out by the mathematician in the usual

STAT

way. The working procedure would be planned according to the old methods of numerical calculation, and the solution of the problem with the automatic calculator would consist only in the calculation of terms in advance of their fixed positions in a planned and known sequence.

The first "automatic calculators" were the result of constructions according to this naive description. This was a type of calculating machine, consisting of a combination of machines with which the operator calculated formulas mechanically. The gadget which controlled the sequence of manipulations and operations with the numbers resembled a telegraph transmitter. A group of openings in perforated paper tape was scanned and the place of transmission of the imprint carried the manipulation and operation with the numbers inside the calculator.

The modern automatic computer does not proceed slavishly according to a preliminarily-prepared sequence of manipulating and operating commands, which are called the instructions. The automatic calculator described in the first part of this manual selects while calculating the instructions in dependence on the results of the operations and usually creates regularly new instructions when old planned instructions are absent. The plan according to which the selection of instructions is carried out and the creation of new instructions is called the instruction network. We shall see that such a plan differs substantially from the instruction sequence of the instruction-machine part with scanning of the telewriter tape.

The development of the numerical method of calculating has been greatly influenced by every advance in the art of calculating means. When, however, the influence of logarithmic tables and calculating machines is compared with the influence of automatic calculators, it is seen that the change in the method of numerical calculation was a revolution. Specifically, with the classical method it was sought to decrease the number of numerical operations or to simplify them (e.g., replacing multiplication by addition). The extremely large number of operations made this expedient impracticable. The modern method of automatic calculation

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

disregards the number of operations, so that it can utilize the calculating methods which are impracticable from the viewpoint of the classical method. (For example, the functional value of a trigonometric function is found in the classical method with the help of a Table, while with automatic calculation any such value is easily and advantageously calculated as the sum of the members of the corresponding polynominal). In automatic calculation, a large number of operations is not disadvantageous because its operating speed is considerable.

The application of automatic computing to the solution of technical and research problems has, however, one fundamental trait which should be stressed: every solution of a problem with automatic calculation represents a complex mathematical experiment carried out according to arbitrarily-established conditions. By varying these conditions and comparing the variation in the results it is often possible to find the answers to other accompanying disagreements. The employment of automatic calculation, of course, brings to light purely theoretical questions, which in many cases serve as the starting point for new research in other fields of mathematics.

INTRODUCTION TO THE METHOD OF OPERATION

1.2. Calculation by Formulas

In formulating a working procedure for automatic computing, the start is made with the classical calculating method by formulas. The working procedure begins with the selection of a suitable numerical method, the notation of the mathematical expressions, the analysis and planning of the operations, the selection of the starting values, and the preparation of the formulas. If the starting values are arranged according to a single index (in sequence), the calculating is carried out with the help of the single formula in the form of separate Tables. At a large number of independent variables (large number of indexes) a bound volume of Tables or a whole series of such volumes is prepared.

The formula with the plan of operations constitutes a model of the selected

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

calculating method. An example of a classical formula is shown in Fig.1.1. The last column, the values of g_s are calculated in succession with the help of the formula from the values of s given in the first column according to the expressions given in the headings of the columns. These constitute the instruction series.

The operation procedure when working by formula begins by taking one, two or more of the values written in the formula, proceeding with the operation in accordance with the expressions in the headings of the columns, and finally noting the results in the corresponding places of the formula. The columns are filled with various values of the same variables, while the lines contain the values of various variables which are operationally dependent on each other.

The operations in working by formulas are usually carried out with the calculating machine or with the help of Tables. The formula is usually arranged in columns because the arrangement in line has great disadvantage when using the calculating machine and Tables. On the basis of the columns already filled, the unfilled columns are filled. At the filling of the same column the operation remains unchanged for all of its places, the computer remains set for the same type of operation, the values vary regularly, the setting of the values in the machine and their registration are regular. Thus it is little fatiguing, easily supervised, and therefore reliable.

The working procedure such as filling a formula by columns is suitable for devices operating with perforated plates. It is unsuited as the basis for the plan of operational sequence in automatic calculation, because it makes impossibly high demands on the memorizing capacity of the machine.

1.3. Instructions

In calculating by formula, the instructions constitute the directions for filling further places in the formula. The instructions, therefore, contain the following information:

whence: from which place of the formula to take the number to be operated on:

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

direction: how to operate on the taken number;

where: into which place of the formula to insert the result.

For the working procedure by formula shown in Fig.1.1 a sequence of operations of this type can be prescribed. The number of terms of such a sequence will be equal to the number of operations necessary for filling the formula (counting one operation for each place). By selecting suitable symbols the instructions for the formula can be prepared. While such instructions are never prepared in practice, it seems desirable for the present purpose to explain the working procedure at automatic calculation.

1.4. Instruction Symbols

The address is the number which clearly expresses the place of the formula. The address determines the place from which the number to be operated on is taken, and the place where the result of the operation is noted. The address also denotes the place into which the instruction is written. If it is not expressly desired to state the content of a place, i.e., whether by number or instruction, the expression "word" is used for the content of the place.

The number assigned to the address a is denoted by the symbol <a>.

The instruction assigned to the address b is denoted by the symbol <b>.

The equation x = <a> means either of the following two expressions: "x = the number assigned to the address a", or "the number assigned to the address A = x".

No developing character is attributed to the equation. The equation is simply a verifying equality.

Operations: An operation here is every fully defined working procedure by which a new item of information is obtained from a previous item of information. Arithmetical operations are denoted by the usual symbols. If it is desired to express the value to be operated by a given address the simple symbol < > is used as defined in the preceding paragraph.

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

s	$10^{s/10} - 1 = X_1$	$150° - 0.111° \cdot X_1 = X_1^{(0)}$	$\sin X_1^{(0)}$	$\cos X_1^{(0)}$	$0.1 \cdot s$	$-116.5° + 90° \cdot 0.1 \cdot s = X_2^{(0)}$	$\sin X_2^{(0)}$	$\cos X_2^{(0)}$	$0.707 \cdot \sin X_1^{(0)} - 0.55 \cdot \cos X_2^{(0)} = V_x$	$1 - 0.707 \cdot \cos X_1^{(0)} - 0.55 \cdot \sin X_2^{(0)} = V_y$	$V_x^2 + V_y^2 = \eta$
0 0	20 0.00000	40 150.000	60 0.50000	80 −0.86603	100 0.0	120 −116.50	140 −0.89493	160 −0.44620	180 0.84571	200 0.63313	220 1.11608
1 1	21 0.25892	41 148.418	61 0.52371	81 −0.85180	101 0.1	121 −107.50	141 −0.95372	161 −0.30071	181 0.89481	201 0.56310	221 1.11777
2 2	22 0.58489	42 146.426	62 0.55306	82 −0.83317	102 0.2	122 −98.50	142 −0.98902	162 −0.14781	182 0.93491	202 0.49224	222 1.11647
3 3	23 0.99526	43 143.917	63 0.58901	83 −0.80817	103 0.3	123 −89.50	143 −0.99996	163 0.00873	183 0.96641	203 0.42382	223 1.11357
4 4	24 1.51189	44 140.761	64 0.63236	84 −0.77451	104 0.4	124 −80.50	144 −0.98629	164 0.16505	184 0.98968	204 0.36161	224 1.11025
5 5	25 2.16228	45 136.786	65 0.68472	85 −0.72880	105 0.5	125 −71.50	145 −0.94832	165 0.31730	185 1.00567	205 0.31022	225 1.10761
6 6	26 2.98107	46 131.782	66 0.74568	86 −0.66630	106 0.6	126 −62.50	146 −0.88701	166 0.46175	186 1.01505	206 0.27406	226 1.10593
7 7	27 4.01187	47 125.483	67 0.81428	87 −0.58046	107 0.7	127 −53.50	147 −0.80386	167 0.59482	187 1.01782	207 0.26246	227 1.10484
8 8	28 5.30957	48 117.552	68 0.88659	88 −0.46255	108 0.8	128 −44.50	148 −0.70091	168 0.71325	188 1.01232	208 0.28009	228 1.10358
9 9	29 6.94328	49 107.569	69 0.95336	89 −0.30185	109 0.9	129 −35.50	149 −0.58070	169 0.81412	189 0.99341	209 0.33883	229 1.10167
10 10	30 9.00000	50 95.000	70 0.99619	90 −0.08716	110 1.0	130 −26.50	150 −0.44620	170 0.89493	190 0.94972	210 0.44617	230 1.10104

Fig.1.1 - Calculation by Formula

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

Thus, for example:

$\langle a \rangle + \langle b \rangle$ ... denotes addition of the numbers assigned to addresses a and b.

$\sin \langle p \rangle$ ... denotes the sine of the argument assigned to address p;

$\langle m \rangle + n$ ... denotes the addition of the number n to the number assigned to the address m;

$m + n$ ... denotes the addition of m to n;

$\langle 2 \rangle \times \langle 3 \rangle$ ... denotes the product of the numbers assigned to the addresses 2 and 3.

Development: The result of the temporal development of the working procedure is denoted by the development-sign $\rightarrow$, at which:

$x \rightarrow \langle a \rangle$ ... denotes the number x substituting the number assigned to address a;

$\langle a \rangle \rightarrow \langle b \rangle$ ... denotes the number assigned to address "a" substituting the number assigned to address "b";

$\langle c \rangle \rightarrow y$ ... denotes the number assigned to address "c" substituting the number y;

$x \rightarrow y$ ... denotes the number x substituting the number y.

$\langle a \rangle + \langle b \rangle \rightarrow \langle c \rangle$ ... denotes the addition of the numbers assigned to addresses a, b, substituting the number assigned to address c;

$n + 2 \rightarrow n$ ... denotes the addition n + 2 substituting the number n;

$\langle a \rangle + \langle a \rangle \rightarrow \langle a \rangle$ ... denotes the addition of the number assigned to address a to the (same) number assigned to address "a" substituting for the number assigned to address "a".

The last expression illustrates the intermediary character of the working procedure: On the address "a" the number $\langle a \rangle$ is taken twice. The addition of $\langle a \rangle + \langle a \rangle = 2\langle a \rangle$. This result is assigned back to address "a". After carrying out the

STAT

instruction, it is obvious that the number on address "a" will be twice what it was before. The development sign must never be substituted by the equal sign.

The symbols connected by the development sign constitute the operation symbol. To the left of the development sign of this symbol is the operation prescribed by the operation sign, whose execution gives the resultant information. The development sign symbolizes the working procedure, in which the originally filled right side of the operation symbol with the development change in the resultant information is defined by the left side.

The unfilled instruction symbols just defined are sufficient for specifying the working procedure for filling the formula of Fig.1.1. This specification is briefly set forth in Fig.1.2 in the form of the sequence of operation symbols. Such concrete calculating instructions do not have much practical value for a calculation by formulas, and its mechanization leads to the conception of automatic calculation of an obsolescent type.

A cursory glance at Fig.1.2 shows that some of the operation symbols have a common form, although the contents differ. The necessity may arise of carrying out an operation for changing the content of an instruction. It may also sometimes be desirable to repeat a same instruction two or more times. All that is necessary for doing this is to select between the instructions as required during their execution.

If after the end of an operation this latter is continued according to an instruction selected from a number of possibilities, the possibility is obtained of ramifying the working procedure in dependence on the result of the final operation. If one of the branches which has been formed in this way is introduced into an instruction which has already been carried out, an iteration process is obtained. For facilitating the description of the iteration process, the instruction symbols are supplemented by the instruction symbols of several data.

<u>Complete Instructions</u>. The symbols presented above contain the following information: from where to take the number to be operated on, how to carry out the

STAT

operation, where to assign the result. The specification is supplemented with:

The address to which the instruction is assigned so that it can be continued if the result of the operation is negative.

The address to which the instruction is assigned so that it can be continued

$$
\begin{array}{lcl}
10^{\langle 0\rangle/10} - 1 & \rightarrow & \langle 20\rangle \\
10^{\langle 1\rangle/10} - 1 & \rightarrow & \langle 21\rangle \\
\cdots & & \cdot \\
\cdots & & \cdot \\
\cdots & & \cdot \\
10^{\langle 10\rangle/10} - 1 & \rightarrow & \langle 30\rangle \\
150^\circ - 6.111^\circ . \langle 20\rangle & \rightarrow & \langle 40\rangle \\
150^\circ - 6.111^\circ . \langle 21\rangle & \rightarrow & \langle 41\rangle \\
\cdots & & \cdot \\
\cdots & & \cdot \\
\cdots & & \cdot \\
150^\circ - 6.111^\circ . \langle 30\rangle & \rightarrow & \langle 50\rangle \\
\sin \langle 40\rangle & \rightarrow & \langle 60\rangle \\
\cdots & & \cdot \\
\cdots & & \cdot \\
\cdot & & \cdot \\
\cdot & & \cdot \\
\cdot & & \cdot \\
\langle 180\rangle^2 + \langle 200\rangle^2 & \rightarrow & \langle 220\rangle \\
\langle 181\rangle^2 + \langle 201\rangle^2 & \rightarrow & \langle 221\rangle \\
\cdots & & \cdot \\
\cdots & & \cdot \\
\cdots & & \cdot \\
\langle 190\rangle^2 + \langle 210\rangle^2 & \rightarrow & \langle 230\rangle
\end{array}
$$

Fig.1.2 - Instruction Sequence of Expressed Operation Symbols

if the result of the operation is positive.

The use of two addresses in each of the instructions assumes that each of the instructions has its own address, at which the specification is revoked. Complete general instruction symbols are assigned to a formula according to the following model:

The first column contains the address to which the instruction is assigned, with the specified symbol on the same line to the right. The second column is the

STAT

operation symbol. The third column is the address to which the instruction is assigned if it is to be continued in the event that the result of the operation is not negative. The fourth column represents the address to which the instruction is assigned if it is to be continued in the event that the result is negative.

		+	−
AA	$\langle a \rangle + 1 \rightarrow \langle a \rangle$	AB	AB
AB	$\langle a \rangle - N \rightarrow \langle b \rangle$	AC	AA
AC	$0 \rightarrow \langle a \rangle$	AD	AD

Fig.1.3 - Group of Complete Instructions in General Form

If the addresses in two sucessive columns are equal to each other (as is the case with instruction! <AA> and ! <AC>, the operation is continued regardless of the sign for the result obtained in the operation. With instruction ! <AB>, instruction ! <AA> is repeated until the number on address "a" has increased at least to N. Then the operation is continued according to instruction ! <AC>.

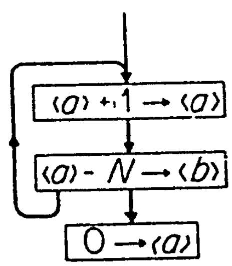


Fig.1.4 - Development Diagram Corresponding to Group of Instructions of Fig.1.3

The addresses of the instructions in the general instruction symbols are usually denoted by two capital letters.

The instruction network is completed by the logical joining of the system of instructions according to which the given problem can be solved.

The working procedure is often represented in the form of the development diagram, in which the instructions are expressed by graphic operation symbols entered in the fields. The fields are joined by arrows indicating the development of the working procedure and its ramification in dependence on the sign of the result of the operation. A simple example of such a diagram is shown in Fig.1.4. In complicated cases it resembles a network with many branches, and is appropriately known

as the instruction network.

The working procedures specified in Figs.1.3 and 1.4 are concordant.

1.5 Example for an Instruction Network

The preparation of the instructions is carried out first purely by formula, in order not to have to include new terms, which might give rise to difficulties.

a)	b)			Instruction	+	−
	0 ⟨0⟩		!Q0			
	1 ⟨1⟩		!Q1			
	2 ⟨2⟩		!Q2			
	X 1,44	3	!Q3			
	ε 0,001	4				
	x_n	5	AA	⟨1⟩ → 5	AB	AB
$x_0 = 1$	$\frac{X}{x_n}$	6	AB	⟨3⟩ : ⟨5⟩ → 6	AC	AC
$X:x_n$			AC	⟨6⟩ − ⟨5⟩ → 7	AD	AD
$\left\lvert\frac{X}{x_n} - x_n\right\rvert$ q	q	7	AD	⟨7⟩ − ⟨4⟩ → 7	AE	$\boxed{AG}$
$q - \varepsilon$	$q - \varepsilon$	7	AE	⟨6⟩ + ⟨5⟩ → ⟨7⟩	AF	AF
$\frac{X}{x_n} + x_n$ $2x_{n+1}$	$2x_{n+1}$	7	AF	⟨7⟩ : ⟨2⟩ → ⟨5⟩	$\boxed{AB}$	$\boxed{AB}$
$2x_{n+1}:2$ x_{n+1}	$x_{n+1} \to x_n$		AG			
c)						

Fig.1.5 – Instruction Network for Root Extraction

a) Mathematical expressions (Rozbor); b) Address (Slovnik)

c) Stop, perforation

Let us calculate the root of 1.44 by the iterative process, defined by the relation

$$x_{n+1} = \frac{1}{2}\left(\frac{X}{x_n} + x_n\right)$$

for n = 0.1,..., x = 1, X = 1.44.

The condition for continuing with iteration is

$$\left|\frac{X}{x_n} - x_n\right| = q \geq \varepsilon = 0{,}001 .$$

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

The result is regarded as the final value of x_n.

Procedure at a Proposed Instruction Network

First select the address to which the constant is assigned. Then form in succession the values expressing the variables. First calculate q which furnishes the criterion for continuing with iteration. If q is not smaller than e, calculate $x_n + 1$, repeat the calculation of q, and according to its value continue with iteration, or terminate the iteration and enter the result on the card. A working specification of a proposed instruction network is given in Fig.1.5.

Investigation of the working procedure given by this instruction network prepared by the formula shows that a total of 8 fields (addresses) is sufficient. At actual calculation, however, it is often necessary to specify names in several fields. An example of the changes in this formula is presented in Fig.1.6. In the sixteen columns of this Table, the sequence of the development of the whole formula is shown according to the instructions denoted by the addresses in the headings.

AUTOMATIC CALCULATION

1.6. Simple Scheme

The working procedure of automatic calculation is illustrated in the form of a very simple scheme in Fig.1.7. The principal parts of the automatic calculator are shown:

- the memory - representing the formula;
- the control - representing the calculator;
- the operational units - representing the Table and the computer;
- the entrance - serving for feeding the initial information;
- the exit - serving for reading off the result.

1.7. Working Procedure

Before starting to calculate, the prepared batch of perforated cards containing the initial numerical data and instructions is placed in the inlet. The automatic

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

a)

		AA	AB	AC	AD	AE	AF	AB
0	0	0	0	0	0	0	0	0
1	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
2	2,00000	2,00000	2,00000	2,00000	2,00000	2,00000	2,00000	2,00000
3	1,44000	1,44000	1,44000	1,44000	1,44000	1,44000	1,44000	1,44000
4	0,00100	0,00100	0,00100	0,00100	0,00100	0,00100	0,00100	0,00100
5		1,00000	1,00000	1,00000	1,00000	1,00000	1,22000	1,22000
6			1,44000	1,44000	1,44000	1,44000	1,44000	1,18033
7				0,44000	0,43900	2,44000	2,44000	2,44000

	AC	AD	AE	AF	AB	AC	AD	AC
0	0	0	0	0	0	0	0	0
1	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
2	2,00000	2,00000	2,00000	2,00000	2,00000	2,00000	2,00000	2,00000
3	1,44000	1,44000	1,44000	1,44000	1,44000	1,44000	1,44000	1,44000
4	0,00100	0,00100	0,00100	0,00100	0,00100	0,00100	0,00100	0,00100
5	1,22000	1,22000	1,22000	1,20016	1,20016	1,20016	1,20016	$\boxed{1,20016}$ → b)
6	1,18033	1,18033	1,18033	1,18033	1,19984	1,19984	1,19984	1,19984
7	→ 0,03967	0,03867	2,40033	2,40033	2,40033	0,00033	—0,00067	—0,00067

Fig.1.6 - Film Strip of Contents of Memory at Root Extraction According to Instruction Network of Fig.1.5

a) Address instruction; b) Result

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

computer reads off the information on the cards and assigns it to its memory. The memory, which represents the formula, stores the initial data. After starting the machine, the control selects from the memory the first instruction, composed of the operation sign and five addresses i, j, k, r, s. The control selects the numbers from the memory of addresses i, j, and sends them to the operation unit. Simultaneously, the control acts on the operation unit in such a way as to carry out the operation specified by the sign f. The result of the operation is assigned by the control to the memory of address k, where the sign of this result is read off. After setting the read-off sign, the control selects the next instruction from the memory of address r (if the result is positive), or from the memory of address s (if the result is negative).

The automatic computer proceeds in this way from instruction to instruction, pursuing various paths in the instruction network, until it arrives at the instruction which gives the command for stopping the machine. The result of the calculation is then already prepared in the memory and recorded on the batch of perforated cards prepared in the exit.

A detailed description of all of the operations which the automatic computer carries out is given in Chapter 2. It should be noted that the combined operations can be extended for carrying out any instructions. Operations are not only the fundamental arithmetical operations with numbers, but also the working process by which the machine reforms or forms instructions or numbers. By such operations, for example, the machine changes the instructions of any of five addresses i, j, k, r, s and the operation sign f.

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

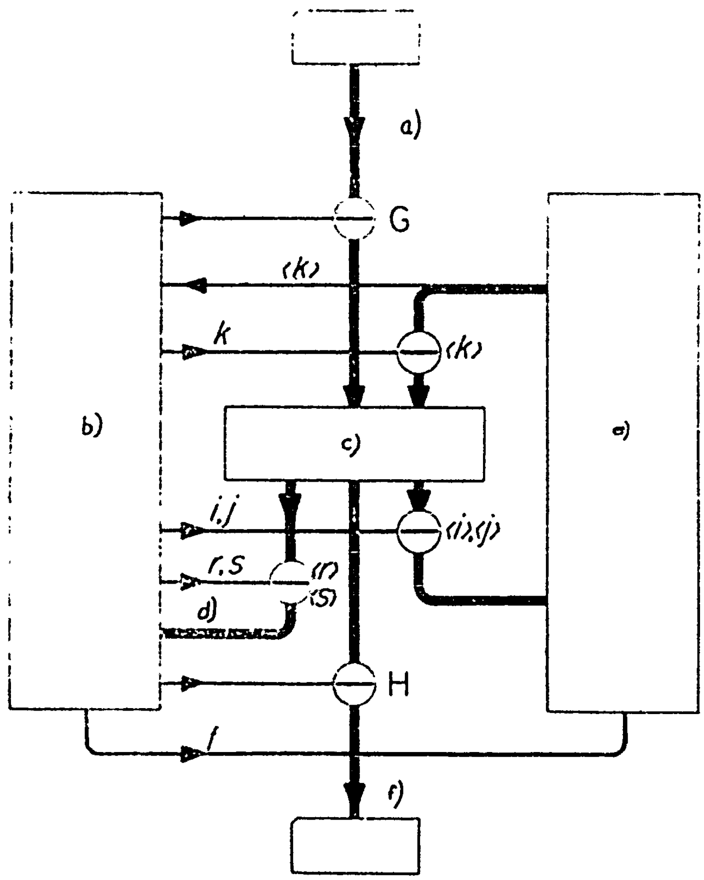


Fig.1.7 - Idealized Scheme of an Automatic Computer

a) Entrance; b) control; c) Memory;

d) Instructions; e) Operational unit; f) Exit

[redacted]

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

CHAPTER 2

CODES OF AUTOMATIC CALCULATOR

WORDS

2.1. Words

The automatic computer processes information according to words, sometimes created, of invariable magnitude. Each work is composed of 32 binary numbers, which just fills one place in the memory (denoted by one address). Of these 32 numbers, 31 are carriers of specific information, while the remaining number is formed in dependence on all the other numbers of the word in such a way that the sum of all 32 numbers of the word is odd. The dependent number is called "parity". If it is denoted by a question mark and the other numbers by dots, we get the picture of the word

? ....

The parity serves for verifying the correctness of the transfer of the word to the machine.

By definition, the number or instruction is the word. The specification by which the number or instruction is expressed by a word is called the code.

CODES OF NUMBERS

Our automatic computer employs for the depiction of numbers two different codes: Code B and Code D. If the number is given in the binary form, it can be directly depicted by Code B, while if the number is given in the decadic form it can be directly depicted by Code D.

2.2. Code B

The automatic computer works with the binary number N given on 24 valid binary numbers. The highest valid digit of such a number may have various orders of

STAT

magnitude depending on the position of the binary decimal point. It is very advantageous to place the binary decimal point in such a way that it stands close to the left ahead of the highest valid digit of the given number, and so that a new number is formed whose product of multiplication by the corresponding power of 2 is equal to the original number.

For example, the number

IIIIOOOOIIII,OOOOIIIIIII

Is in this way transformed to the product of

$$,IIIIOOOOIIIIOOOOIIIIIII \, . \, 2^{12}.$$

The number formed by this transformation is called the "binary numerical picture X" of the number N. The power of two is called the "Exponent P". The absolute value of the given number N is then expressed by the double number X, P, at which there is valid

$$|N| = X \cdot 2^P$$

and at the same time

$$2^{-1} \leq X < 2^0$$

The symbol of the number N is depicted by the binary digit Z. At positive N Z = 0, at negative N Z = I.

The absolute value of the exponent is expressed by a five-place binary number, the symbol of the exponent is expressed by the binary digit z according to the same specification as with the symbol of the number N.

The word depicting the number N according to Code B is formed according to the example

?	,I......................	z	.....	Z
parity	binary numerical picture of X	symbol	absolute value	symbol for number N
		exponent		

Before the particular word depicting zero is introduced, the latter cannot be expressed at all because the first digit of a nonzero binary numerical picture is

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

always I. Zero is expressed by a word according to the example

I ,000000000000000000000000 0 00000 0 .

This example is common to both codes B and D*.

2.3. Code D.

The automatic calculator receives and delivers decadic numbers N with 6 valid decadic digits. As with Code B, the decimal point is placed in such a way that it stands to the left ahead of the first valid digit, and so that the formed number is multiplied by the corresponding power of ten. The absolute value of N is then expressed by the double number T, Q at which is valid:

$$|N| = Y \cdot 10^{Q}$$

and simultaneously

$$10^{-1} \leqq Y < 10^{0}.$$

The symbol of the number N is expressed by the digit Z, and the symbol of the exponent Q is expressed by the digit z in exactly the same way as in Code B.

Six decadic digits of a decadic numerical picture Y are expressed by six four-place binary numbers according to the model.

, ...

The absolute value of the exponent Q is expressed by a five-place binary number, whose highest digit is always zero.

---

* The binary numerical representation is denoted with the corresponding octet in such a way that the triplets of the binary number are expressed in succession by the octic digits 0, 1, 2, 3,,4, 5, 6, 7. For example, the binary numerical representation,

,III IOO OOI III OOO OII III III

is denoted in the form of

,74170377.

In Code B, the exponent P is also affixed to the octet.

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

The word depicting the number N according to Code D has the form

?	, ...	z	0....	Z
parity	decadic number of picture Y	symbol	absolute value	symbol for number N
		exponent		

For example, the number - 45,0769 is expressed according to Code D by the word

I , OIOO OIOI OOOO OIII OIIO IOOI O OOOIO I

2.4. Numerical Range of Machine

The numerical range of the machine is determined by Code B and not by Code D. Therefore the automatic calculator calculates only with the numbers depicted according to Code B. A number fed to the machine in Code D must be transformed to Code B before it is used. From this it follows that the machine is unable to process numbers higher than the highest of all the numbers expressed according to Code B.

The highest number depicted according to Code B is

$$2\,147\,483\,520 \quad 2^{31}.$$

This number is depicted according to Code B by the word

O ,IIIIIIIIIIIIIIIIIIIIII O IIIII O.

The smallest nonzero, positive number depicted according to Code B is

$$0{,}000\,000\,000\,232\,831 = 2^{-32}$$

Its depiction is

O ,IOOOOOOOOOOOOOOOOOOOOOO I IIIII O.

Still smaller numbers are depicted by the machine as zero.

CODES OF INSTRUCTIONS

2.5. Codes of Instructions

An instruction contains the directions for the operation and five addresses (see Sect.1.4). It is composed of two words, each of which contains, besides the parity, three binary numbers A, B, C. The number A has 10 binary digits, the B has

STAT

9, the C has 12. The word depicting one half of the instruction has the form

?	....	....	....
parity	A	B	C

The double forming the single instruction is always fed to two neighboring places of the memory. The first part of the instruction is always fed to the memory of even address. The other part of the instruction as a rule is fed to the memory of the next higher address. To prepare the machine with memory of the instruction for the operation, select the first half of the instruction according to the even address.

The instruction (both words) contains the operational signs f_1 and f_2 and 5 addresses i, j, k, r, s.

In the first word,,assigned to the even address:

A .... denotes a ten-place binary number given by the address k of the place in the memory to which the result of the operation, denoted $\langle k \rangle$, is to be assigned;

B .... denotes a nine-place binary number given by the even address (so that 9 places are sufficient) r of the instruction according to which the machine will continue to calculate in the event of a positive result of the operation ($\langle k \rangle \geqq 0$);

C .... denotes a twelve-place binary number f_1 given according to the operational code of the operation which the machine has to carry out according to the words fed to the memory of addresses i, j. The individual digits of the operational sign f_1 are denoted in the form

LJKNDSWXYZMT

In the second word, assigned to the address by one higher than the address of the first word:

A .... denotes a ten-place binary number given by the address j of the number $\langle j \rangle$ to be operated on;

B .... denotes a nine-place binary number given by the even address of the

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

instruction according to which the machine will continue to calculate in the event of a negative result of the operation ($\langle k\rangle < 0$);

C .... denotes a twelve-place binary number composed of the two-place operational sign f_2 and a ten-place binary number given by the address i of the number $\langle i\rangle$ to be operated on. The digits of the operational sign f_2 are given according to the operational code of the action of the entrance and exit units of the machine. It is designed in the form of

GH..........

	Parity	A	B	C
1. Word	?	....	....	....
		k	r/2	f_1
2. Word	?	....	....	$\underbrace{..}_{f_2}\underbrace{..........}_{i}$

Fig.2.1 - Arrangement of Information in Words Constituting the Instruction

2.6. Operational Codes

The selection of the operation is carried out by the machine according to the operational signs f_1, f_2. Here the binary digits of both operational signs are combined to one fourteen-place sign in the form of

IJKNDSWXYZMTGH

regardless of the fact that the part GH is in another part of the instruction than the remaining part IJKNDSWXYZMT. The combined sign is here more legible. Included in the instruction are only those elements of the operational sign f which correspond to places where the sign of unity is in the binary picture. For example, if the binary picture of the operational sign is

IJKNDSWXYZMTGH = 0000I000000I00 ,

the instruction is abbreviated to

DT.

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

The operational code, according to its operation, is depicted by the operational sign f. A survey of all of the operational signs is presented in the Tables of Figs.2.2 - 2.5. In the first column of the Tables there are symbols in general form, in the third column are the detailed denotations of both words specified by the pertinent instructions according to the instruction code. The double capital letters by which the addresses are denoted are only included for convenience, and have no relation to the operations.

MUTUAL RELATIONS BETWEEN OPERATIONAL SIGNS

The meanings of the operational signs are given in Figs.2.2 - 2.5. It is, however, necessary to discuss in detail the operations and the mutual relations between the operational signs.

2.7. Principal Operational Signs

Sign: S ... Addition with correction $\langle i \rangle + \langle j \rangle \rightarrow \langle k \rangle$

K ... Multiplication with correction $\langle i \rangle \cdot \langle j \rangle \rightarrow \langle k \rangle$

D ... Division with correction $\langle i \rangle : \langle j \rangle \rightarrow \langle k \rangle$

The machine starts here with $\langle i \rangle$, $\langle j \rangle$ from the memory of addresses i, j. The specification of the operation is carried out with respect to the sign. The result of the operation is rounded off to 24 valid binary digits. The corrected result replaces the memory of address k. The next instruction is selected according to the sign of the result.

Sign: T ... transformation T1 $\langle i \rangle \rightarrow \langle k \rangle$ at $j = 0$

T2 $\langle i \rangle \rightarrow \langle k \rangle$ at $i = 0$

At $j = 0$, the machine carries out the transformation of T1 to the number $\langle i \rangle$. The machine starts with the memory of the address i of the number depicted according to Code D. The picture of the same number is formed according to B and the result replaces the content of the memory of address k. The machine requires that the exponent Q of the number $\langle i \rangle$ be equal to zero.

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

At i = 0, the machine carries out the transformation of Ta to the number ⟨j⟩. The machine starts with the memory of the address j of the number depicted according to Code B. There is formed the picture of the same number according to Code D and the result replaces the memory of address k. The machine requires that the number ⟨j⟩ be corrected in advance in such a way that $0.1 \leq \langle j \rangle < 1$.

Sign: ST ... addition without correction $\langle i \rangle \underline{+} \langle j \rangle \rightarrow \langle k \rangle$

NT ... multiplication without correction $\langle i \rangle \underline{\cdot} \langle j \rangle \rightarrow \langle k \rangle$

DT ... division without correction $\langle i \rangle \underline{:} \langle j \rangle \rightarrow \langle k \rangle$

The machine starts with the numbers ⟨i⟩, ⟨j⟩ from the memory of addresses i, j. The specification of the operation is carried out with respect to the sign. The result of the operation is handled in such a way that the first 24 valid binary digits are left without correction. The result of the operation replaces the memory of address k. The next instruction is selected according to the sign of the result.

Symbol: SX ... elimination of the exponent $\text{Exp}\ \langle j \rangle + \langle i \rangle \rightarrow \langle k \rangle$

The machine starts with the numbers from the memory of addresses i, j. From the number ⟨j⟩ the exponent P or Q is eliminated. This exponent is annexed to the number ⟨i⟩ with respect to the sign of the result depicted according to Code B, replacing the content of the memory of address k. According to the sign of the result the next instruction is selected.

Symbol: WX ... replacement of the exponent $\langle i \rangle = \text{Exp}\ \langle j \rangle \rightarrow \langle k \rangle$

The machine starts with the numbers from the memory of addresses i, j. From the number ⟨j⟩ the exponent P or Q is eliminated, which is replaced by the number ⟨i⟩. The result replaces the content of the memory of address k. According to the sign of the result, the next instruction is selected.

Symbol: SYZ ... elimination of the sign $\text{Sgn}\ \langle j \rangle \rightarrow \langle k \rangle$

The machine starts with the number from the memory of address j. From this number is eliminated the symbol of the digit Z. This digit as a number according to Code B, and replaces the content of the memory of address k.

Symbol: WYZ ... replacement of the symbol

The machine starts with the numbers from the memory of address i, j. From the number ⟨j⟩ is eliminated the symbol Z and replaced by the number ⟨i⟩. The result replaces the content of the memory of address k.

Symbol:	SXY ... elimination of the number A	A!	$\langle j\rangle + \langle i\rangle \rightarrow \langle k\rangle$
	SZ ... elimination of the number B	B!	$\langle j\rangle + \langle i\rangle \rightarrow \langle k\rangle$
	SY ... elimination of the number C	C!	$\langle j\rangle + \langle i\rangle \rightarrow \langle k\rangle$

The machine starts with the content from the memory of address j (which is one half of the instructions) and the number from the memory of address i. From the word !⟨j⟩ the given number (A, B or C) is eliminated, which is annexed to the number ⟨i⟩ with respect to the sign and the result, which is depicted according to Code B, replacing the content of the memory of address k. According to the sign of the result the next instruction is selected (see example in Fig.2.6).

Symbol:	SWXY ... annexation to the number A in the instruction	$A!\langle j\rangle + \langle i\rangle \rightarrow !\langle k\rangle$
	SWZ ... annexation to the number B in the instruction	$B!\langle j\rangle + \langle i\rangle \rightarrow !\langle k\rangle$
	SWY ... annexation to the number C in the instruction	$C!\langle j\rangle + \langle i\rangle \rightarrow !\langle k\rangle$

The machine starts with the content from the memory of address j (which is one half of the instruction) and the number z from the memory of address i. In the word !⟨j⟩ the given number (A, B or C) is annexed, the number ⟨i⟩ with respect to the sign. The resulting word (which is changed by one half of the instruction) replaces the content of the memory of address k (see example in Fig.2.7).

Symbol:	WXY ... replacement of the number A in the instruction	$\langle i\rangle = A!\langle j\rangle \rightarrow !\langle k\rangle$
	WZ ... replacement of the number B in the instruction	$\langle i\rangle = B!\langle j\rangle \rightarrow !\langle k\rangle$

WY ... replacement of the number C in the instruction $\quad \langle i \rangle = C\langle j \rangle \rightarrow\, !k$

The machine starts with the content (one half of the instruction) from the memory of address j and the number from the memory of address j. In the word $!\langle j \rangle$, the given number (A, B or C) is eliminated and replaced by the number $\langle i \rangle$. The resulting word (which is changed by one half of the instruction) replaces the content of the memory of address k.

2.8. Supplementary Operational Symbols

Besides the principal operational symbols, which have an independent operational significance and cannot be combined with each other, there also exist supplementary operational symbols, which never occur independently but only in combination with the principal operational symbols.

Symbol: I ... suppression of the sign of the number $\langle i \rangle$ $\quad |\langle i \rangle|$

J ... suppression of the sign of the number $\langle j \rangle$ $\quad |\langle j \rangle|$

At the selection from the memory of addresses i, j the machine suppresses the signs of the digits of the given number and carries out the operation specified by the principal operational symbol. The result of the operation replaces the content of the memory of address k. These symbols do not exclude each other, and can be used in combination with all of the principal operational symbols with the condition that the suppression of the signs of the digits is carried out with the number and not with the instruction.

Symbol: K ... suppression of the sign of the result $\quad |\ldots\ldots| \rightarrow \langle k \rangle$

At the replacement of the content of the memory of address k with the result of the operation given by the principal operational symbol, this result is provided with the positive sign. This symbol can be used in combination with all of the principal symbols with the condition that the result of the operation is a number and not an instruction.

Symbol: M ... minus with the number $\langle i \rangle$ $\quad -\langle i \rangle$

The machine has in all two possibilities of influencing the symbol of the number selected from the memory of address i. At the employment of the supplementary symbol I, the sign is suppressed directly at the selection of the number ⟨i⟩ from the memory, which leads to operation on the positive number. At the employment of the symbol M, the machine selects the sign of the number led to the operation from the memory of address i. Consider now the consequence of the influence on the sign at operation on the number - ⟨i⟩ at combination of the two symbols I and M. The symbol M can be used in combination with all of the principal and supplementary symbols with the condition that ⟨i⟩ is a number and not an instruction.

Symbol: G ... reading the card G ....

The symbol contained in any instruction effects that the machine first reads off the word on the card fed into the receiving end of the machine and assigns it to the memory whose address is perforated in the same card. Then the operations specified by the remaining operational symbols are carried out. If the fed cards contain the corresponding holes, the machine also reads off the successive cards in the same way. The process is continued until the machine encounters a card without the symbol. This symbol G can be used in combination with all of the principal and supplementary symbols except the symbol H.

Symbol: H ... perforation of the card H1 .... from the memory
H2 .... from the operation

The differentiation between the symbols H1 and H2 is carried out by setting the switch on the control board before the computing begins.

The symbol H1 contained in any instruction effects that the machine first perforates into the card the word selected from the memory whose address is pre-perforated in the same card. Then there are carried out the operations specified by the other operational symbols. If the fed cards contain the corresponding holes, the machine also perforates the successive cards in the same way. The process is repeated until the machine encounters a card without the symbol. This symbol can be

employed in combination with all of the principal and supplementary symbols except the symbol G.

The symbol H2 contained in any instruction effects that the machine perforates the card with the results of the operations given by the remaining operational symbols. The result of the operation replaces the content of the memory of address k. The machine at the same time perforates the card with the corresponding index. This symbol can be used in combination with all of the principal and supplementary symbols.

Symbol: (empty) ..... stopping the machine STOP

The machine stops.

a)	b) c)	b) d)	b) +	b) —	e)			
S	*AA*	$\langle i\rangle + \langle j\rangle \rightarrow \langle k\rangle$	*AB*	*AC*	*AA*	*k*	*AB/2*	*S*
						j	*AC/2*	*i*
N	*BA*	$\langle i\rangle \cdot \langle j\rangle \rightarrow \langle k\rangle$	*BB*	*BC*	*BA*	*k*	*BB/2*	*N*
						j	*BC/2*	*i*
D	*CA*	$\langle i\rangle \cdot \langle j\rangle \rightarrow \langle k\rangle$	*CB*	*CC*	*CA*	*k*	*CB/2*	*D*
						j	*CC/2*	*i*
T	*DA*	$T1\ \langle i\rangle \rightarrow \langle k\rangle$	*DB*	*DC*	*DA*	*k*	*DB/2*	*T*
						0	*DC/2*	*i*
	DD	$T2\ \langle j\rangle \rightarrow \langle k\rangle$	*DE*	*DF*	*DD*	*k*	*DE/2*	*T*
						j	*DF/2*	0
ST	*EA*	$\langle i\rangle \underline{\pm} \langle j\rangle \rightarrow \langle k\rangle$	*EB*	*EC*	*EA*	*k*	*EB/2*	*ST*
						j	*EC/2*	*i*
NT	*ED*	$\langle i\rangle \dot{-} \langle j\rangle \rightarrow \langle k\rangle$	*EE*	*EF*	*ED*	*k*	*EE/2*	*NT*
						j	*EF/2*	*i*

Fig.2.2 - Meanings of Operational Marks

a) Operational mark; b) General instruction symbol; c) Address of instruction;

d) Operational symbol; e) Complete instruction

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

a)	b) c)	b) d)	b) ·	b) —	e)			
DT	*EG*	$\langle i\rangle \pm \langle j\rangle \to \langle k\rangle$	*EH*	*EI*	*EG*	*k*	*EH/2*	*DT*
						j	*EI/2*	*i*
SX	*FA*	$\mathrm{Exp}\langle j\rangle + \langle i\rangle \to \langle k\rangle$	*FB*	*FC*	*FA*	*k*	*FB/2*	*SX*
						j	*FC/2*	*i*
WX	*FD*	$\langle i\rangle \times \mathrm{Exp}\langle j\rangle \to \langle k\rangle$	*FE*	*FF*	*FD*	*k*	*FE/2*	*WX*
						j	*FF/2*	*i*
SYZ	*GA*	$\mathrm{Sgn}\langle j\rangle \to \langle k\rangle$	*GB*	*GB*	*GA*	*k*	*GB/2*	*SYZ*
						j	*GB/2*	0
WYZ	*GD*	$\langle i\rangle \times \mathrm{Sgn}\langle j\rangle \to \langle k\rangle$	*GE*	*GF*	*GD*	*k*	*GE/2*	*WYZ*
						j	*GF/2*	*i*
SXY	*HA*	$A(\langle j\rangle) + \langle i\rangle \to \langle k\rangle$	*HB*	*HC*	*HA*	*k*	*HB/2*	*SXY*
						j	*HC/2*	*i*
SZ	*HD*	$B(\langle j\rangle) + \langle i\rangle \to \langle k\rangle$	*HE*	*HF*	*HD*	*k*	*HE/2*	*SZ*
						j	*HF/2*	*i*
SY	*HG*	$C(\langle j\rangle) + \langle i\rangle \to \langle k\rangle$	*HH*	*HI*	*HG*	*k*	*HH/2*	*SY*
						j	*HI/2*	*i*

Fig.2.3 - Meanings of Operational Marks (continuation)

a) Operational mark; b) General instruction symbol; c) Address of instruction; d) Operational symbol; e) Complete instruction

a)	b) c)	b) d)	b) +	b) −	e)			
SWXY	IA	$A!\langle j\rangle + \langle i\rangle \to !\langle k\rangle$	IB	IB	IA	k	IB/2	SWXY
						j	IB/2	i
SWZ	IC	$B!\langle j\rangle + \langle i\rangle \to !\langle k\rangle$	ID	ID	IC	k	ID/2	SWZ
						j	ID/2	i
SWY	IE	$C!\langle j\rangle + \langle i\rangle \to !\langle k\rangle$	IF	IF	IE	k	IF/2	SWY
						j	IF/2	i
WXY	JA	$\langle i\rangle - A!\langle j\rangle \to !\langle k\rangle$	JB	JB	JA	k	JB/2	WXY
						j	JB/2	i
WZ	JD	$\langle i\rangle - B!\langle j\rangle \to !\langle k\rangle$	JE	JE	JD	k	JE/2	WZ
						j	JE/2	i
WY	JF	$\langle i\rangle - C!\langle j\rangle \to !\langle k\rangle$	JG	JG	JF	k	JG/2	WY
						j	JG/2	i

Fig.2.4 - Meanings of Operational Marks (continuation)

a) Operational mark; b) General instruction symbol; c) Address of instruction;

d) Operational symbol; e) Complete instruction

STAT

a)	b) c)	b) d)	b) +	b) −	c)			
M	KA	... ⟨i⟩ ... → ...	KB	KC	KA	k	KB/2	M...
						j	KC/2	i
I	KD	... |⟨i⟩| ... → ...	KE	KF	KD	k	KE/2	I...
						j	KF/2	i
J	KG	... |⟨j⟩| ... → ⟨k⟩	KH	KI	KG	k	KH/2	J...
						j	KI/2	i
K	KJ	|......| → ⟨k⟩	KK	KK	KJ	k	KK/2	K...
						j	KK/2	i
G	KL	G ...... → ...	KM	KN	KL	k	KM/2	...
						j	KN/2	Gi
H	KO	H1 ...... → ...	KP	KQ	KO	k	KP/2	...
						j	KQ/2	Hi
	KR	H2 ...... → ...	KS	KT	KR	k	KS/2	...
						j	KT/2	..Hi
	KU	STOP			KU	0	0	
						...	...	...

Fig.2.5 - Meanings of Operational Marks (continuation)

a) Operational mark; b) General instruction symbol; c) Address of instruction;

d) Operational symbol; e) Complete instruction; f) Empty

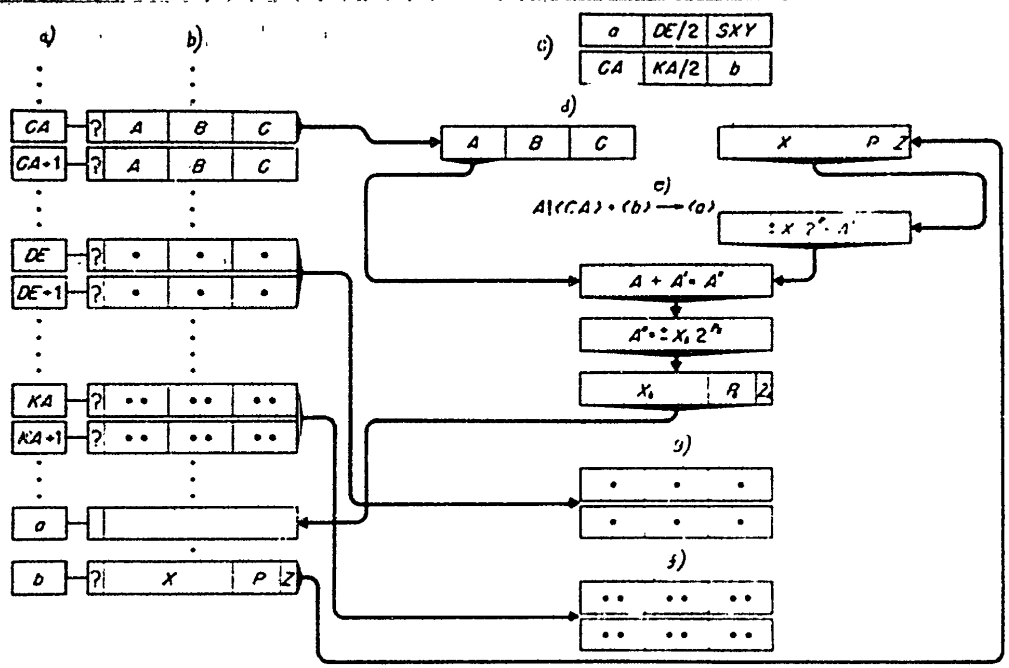


Fig.2.6 - Course of Operations According to Operational Mark SXY

a - Address of memory; b - Content of memory; c - Control receives instruction, d - Machine starts with word (depicted by one half of the instruction) from memory of address CA and word (depicted by number) from memory of address b; e - According to operational mark SXY, machine carries out elimination and addition A'(CA) + (b) → (a), f - Result number assigned to memory of address a; if sign of result is positive ($Z_k = 0$), control selects next instruction from memory of addresses DE, DE + 1; g - If sign of result is negative ($Z_k = 1$), control select next instruction of memory for addresses KA, KA+1

STAT

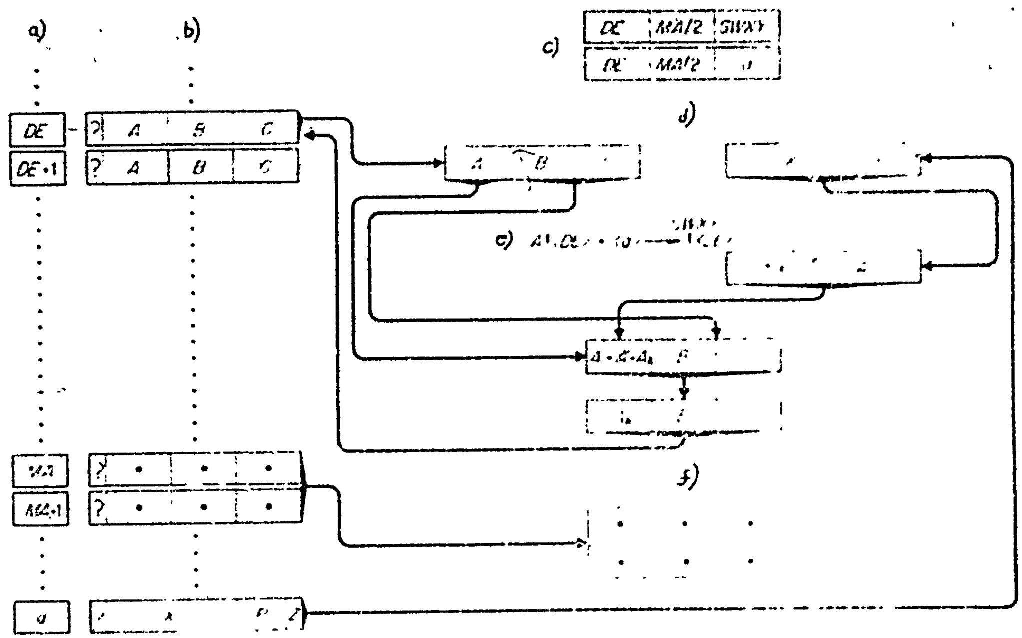


Fig.2.7 - Course of Operations According to Operational Mark SWXY

a) Address of memory; b) Content of memory; c) Control receives instruction; d) Machine starts with word (depicted by half the instruction) from memory of address DE and word (depicted by number) from memory of address a; e) According to operational mark SWXY machine carries out addition of A!(DE) + + (a) ⟶ !(DE); f) The result change in the half instruction replaces the original half instruction in the memory of address DE; control selects next instruction from memory of addresses MA, MA + 1

STAT

2.9. General Survey of Fundamental Operations

Operational Mark	Operational Symbol
S	$\langle i\rangle + \langle j\rangle \rightarrow \langle k\rangle$
N	$\langle i\rangle \cdot \langle j\rangle \rightarrow \langle k\rangle$
D	$\langle i\rangle : \langle j\rangle \rightarrow \langle k\rangle$
T	$T1\langle i\rangle \rightarrow \langle k\rangle$ $T2\langle j\rangle \rightarrow \langle k\rangle$
M	$-\langle i\rangle$
I	$\lvert\langle i\rangle\rvert$
J	$\lvert\langle j\rangle\rvert$
K	$\lvert\ldots\ldots\rvert \rightarrow \langle k\rangle$
G	$G\ldots.$
H	$H1\ldots$ $H2\ldots$
ST	$\langle i\rangle \underline{+} \langle j\rangle \rightarrow \langle k\rangle$
NT	$\langle i\rangle \underline{\cdot} \langle j\rangle \rightarrow \langle k\rangle$
DT	$\langle i\rangle \underline{:} \langle j\rangle \rightarrow \langle k\rangle$
SX	$\mathbf{Exp}\,\langle j\rangle + \langle i\rangle \rightarrow \langle k\rangle$
WX	$\langle i\rangle = \mathbf{Exp}\,\langle j\rangle \rightarrow \langle k\rangle$
SYZ	$\mathbf{Sgn}\,\langle j\rangle \rightarrow \langle k\rangle$
WYZ	$\langle i\rangle = \mathbf{Sgn}\,\langle j\rangle \rightarrow \langle k\rangle$
SXY	$A!\langle j\rangle + \langle i\rangle \rightarrow \langle k\rangle$
SZ	$B!\langle j\rangle + \langle i\rangle \rightarrow \langle k\rangle$
SY	$C!\langle j\rangle + \langle i\rangle \rightarrow \langle k\rangle$
$SWXY$	$A!\langle j\rangle + \langle i\rangle \rightarrow !\langle k\rangle$
SWZ	$B!\langle j\rangle + \langle i\rangle \rightarrow !\langle k\rangle$
SWY	$C!\langle j\rangle + \langle i\rangle \rightarrow !\langle k\rangle$
WXY	$\langle i\rangle = A!\langle j\rangle \rightarrow !\langle k\rangle$
WZ	$\langle i\rangle = B!\langle j\rangle \rightarrow !\langle k\rangle$
WY	$\langle i\rangle = C!\langle j\rangle \rightarrow !\langle k\rangle$
(empty)	STOP

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

CHAPTER 3

THE PREPARATION OF AN INSTRUCTIONAL NETWORK

The procedure for the preparation of an instructional network has three parts, as follows:

a) Selection of a suitable numerical method;

b) Mathematical formulation according to selected method and preparation of the instructional network in general form;

c) Detailed preparation of the instructional network.

WORKING PROCEDURE

3.1. Selection of Numerical Method

The study of numerical methods suitable for automatic calculation is in itself a new branch of mathematics. To this branch, a series of works will be devoted in future yearbooks of the Manual.

3.2. Mathematical Formulation and Preparation of an Instruction Network in General Form

The mathematical formulation of the selected method and the draft instructional network in general form were carried out on form "Model 1", page 1 of which is shown in Fig.3.1.

The three main columns of this form have the headings: "Analysis", "Vocabulary", "Instruction". The "Instruction" column is subdivided into 4 columns, titled: "Index", "Operational symbol", " + ", " – ". The pages of the form are numbered and carry the designation of the job to which they are related. The form is filled with the corresponding mathematical symbols and the symbols described in Chapter 1.

The mathematical formulation begins in the "Analysis" column. The factual data are expressed in succession in mathematical terms, followed by successive specification of the arrangement of these terms. There are also frequently entered mathemat-

ical statements which do not participate directly in the process but constitute the basis thereof. The next step of the construction is the drafting of the instructional network in general form. This is done by filling in parallel the remaining columns of the form. The column "Vocabulary" contains the symbols describing the contents of the addresses and memories. The "Instruction" column has the general instructional symbols according to the mathematical expressions in the "Analysis" column. Each instruction is denoted by an index composed of two capital letters. This is done for facilitating the orientation in the instructional network, since in a group of instructions belonging together the individual instructions all begin with the same initial letter. The second letter gives the order of the instruction in the group (for example, BA denotes the first instruction in the group of instructions B). The column "Instruction" contains also the clews for the constants set in the memory of the machine. A clew has the character of an instruction, and is denoted in the "Index" column by a mark composed of an exclamation mark, the letter Q and an ordinal number (for example, !Q24 is the clew of the number 24 to the stored constant).

3.3. Preparation of the Detailed Instruction Network

The preparation of the detailed instructional network is carried out on "Model 2" form, page 1 of which is shown in Fig.3.2. The four main columns of this form have the headings: "Index", "Address", "Entrance Information", "Variable Information". The "Index" column contains the reference to the line of "Model 1" form. The "Address" column is subdivided into two columns. The first of these columns has preprinted the octonary of the address of the work entered on the same line, and the second contains pre-printed the verifying mark P. The "Entrance Information" is likewise subdivided into two columns. The first of these columns contains all of the words stored in the machine before the beginning to compute, and the second contains the mark symbolizing the numerical code to be used (B or D). The "Variable Information" column gives the symbols of the variables of the constants which occur in the corresponding address during calculation.

STAT

Model 1 Job Page 1

Remarks		
Instruction	-	
	+	
	Operational symbol	
	Index	
Vocabulary		
Analysis		

Fig.3.1 - "Model 1" Form

A complete "Model 2" form consists of 32 pages of 32 lines each, or a total of 1024 lines. The lines of the "Address" column are numbered in advance. Instead of the usual decadic marks the octonary marks from 000 to 1777 are used in the preprint. Each line of the executed form is permanently coordinated with one place of the memory of the machine.

"Model 2" form is filled according to the inserted "Model 1" form. This filling is, of course, quite tedious, and requires concentrated attention, but represents, in contrast to the preparation of the instructional network in the usual form, nearly all mechanical work.

It should be noted that entered on the form at the individual addresses are all the given instructions and constants. For reasons following from the construction of the machine, always inserted into the memory of address 0000 is the constant, and into the memory of the address 0001 the constant 1. These constants are, of course, preprinted on the form. The entering of the instructions is carried out by executing the "Index" column, where the general symbols of the addresses of the instructions are entered according to the "Model 1" form. Thereafter, the constants of the problem set before the beginning to compute are entered. On both forms these constants are denoted by the letter Q. According to the vocabulary of "Model 1" form, the memory is finally replaced by a variable magnitude. On the basis of this work it is advantageous to affix concrete numerical values of the general addresses in the vocabulary. The selection is noted in the "Remarks" column. At the same time, the "Variable Information" column is supplied with the development of the variable content of the memory.

After entering all of the words, the "Entrance Information" column is inserted. The numbers correspond with the employed code (B, D), while the instructions are entered according to the name of the model (Figs.2.2 - 2.5). It must be mentioned that all of the addresses are written in the form of the fundamental octonary.

The preparation of the instructional networks ends with the detailed and com-

STAT

Job

Model 2

Page 1

INDEX	Address		Entrance Information		Variable Information	Remarks
Q0	0000	P	.00000000 0 00 0	B	= 0	
Q1	0001		.40000000 0 01 0	B	= 1	
	0002					
	0003	P				
	0004					
	0005	P				
	0006	P				
	0007					
	0010					
	0011	P				
	0012	P				
	0013					
	0014	P				
	0015					
	0016					
	0017	P				
	0020					
	0021	P				
	0022	P				
	0023					
	0024	P				
	0025					
	0026					
	0027	P				
	0030	P				
	0031					
	0032					
	0033	P				
	0034					
	0035	P				
	0036	P				
	0037					

Fig.3.2 – First Page of "Model 2" Form

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

plete filling of "Model 2" form. According to the "Address" and "Entrance Information" columns there is then perforated the entrance batch of perforated cards.

3.4. Instructional Network for the Calculation of cos x

In the following the preparation of the instructional network for calculating the value of y equals cos x will be described.

Selection of the Method of Calculation

Employed in the calculation is the familiar exponential series

$$\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \cdots$$

Because the value of argument x is unknown in advance, and the series for the magnitude of x converges slowly, it is necessary before proceeding with this series to calculate the argument of the same functional value between 0 and $\pi/2$. The iteration is calculated by addition of the exponential series, because this is the most advantageous method for the automatic computer. The calculation is terminated as soon as the next term of the series is smaller than the given ε.

The mathematical formulation and the drafting of the instructional network in general form are carried out on "Model 1" form shown in Figs.3.3 and 3.4. On the first page of the form, in the "Analysis" column, the arrangement of the argument is formulated in the specified limits. For the sake of simplicity we shall start at this point with the absolute value of the argument. First we subtract 2π until we get a negative result. Then we reach z_N, located in the interval -2π to 0. To this z_N we then add $\pi/2$ until we get a positive result, which will be v_m, located in the range 0 to $\pi/2$. The second column of the Table, entered after completion of the "Analysis" column gives $y = \cos x$ for various possible m values entered in the first column of this Table. Since we wish in principle to calculate only according to the series for the cosine, we put $y = s \times \cos w$. The values for s and w for various m values are given in the remaining two columns of the mentioned Table. On the second

page of the form (Fig.3.4), in the "Analysis" column, the method for calculating the sum of the exponential series is formulated and the condition for its completion is stated.

The drafting of the instructional network for the described arrangement of the argument in general form is carried out in the remaining columns of the form in Figs.3.3 and 3.4. We begin with the two clews !Q0 and !Q1 for setting the standard constants 0 and 1 in the memories 0 and 1 and with the clew !Q3 for setting the given argument x in the memory a. We proceed according to the first instruction AA: "Take the absolute value of the number set in the memory a and replace its content with the same memory a. Regardless of the sign of the result (in this case it will always be positive) continue further according to instruction BB". After carrying out instruction AA, $|x|$ is set in memory a, which is entered in the "Vocabulary" column. The next instruction AB is: "Set the product of the numbers set in memories p1 and pa in the memory b. Proceed further according to instruction AC". After execution of instruction AB, the constant 2π is set in memory b. With this, however, we are prepared to calculate z_n. For the calculation, memory a is used where $z_0 = |x|$ has already been prepared. The calculation is carried out according to instruction AC: "From the content of memory a subtract the content of memory b, and replace the content of memory a by the result. At a positive result proceed further according to instruction, at a negative result proceed further according to instruction AD". It is obvious that the execution of this instruction must be repeated until a positive result is obtained. At the first negative result the operation is discontinued, because then the sought z_N is already set in the memory (and recorded in the "Vocabulary" column), and the calculation is continued according to instruction AD. This instruction is carried out for replacing the content of memory b (where the set constant 2π is no longer needed) by the new constant $\pi/2$. Thus we are prepared to calculate v_m used in memory a. At this point of the instructional network, simultaneously with the calculation of v_m, the ramification of 4 possible

STAT

413

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

further processes specified in the Table in the "Analysis" column for various m values is made. This is carried out using instructions AE, AF, AG and AH. By execution of instruction, the content of memory a or $\pi/2$ is increased. At a positive result we are ready to proceed further, because then m = 1, in memory a the value v_1 is prepared, and we change over to instruction AI, where first w and s at m = 1 are calculated. In the event of a negative result we continue according to instruction AF. The execution of this instruction again increases the content of memory a or $\pi/2$. In the event of a positive result m = 2, we have v_2, and we change over to instruction AK, where first w and s at m = 2 are calculated. In the event of a negative result we continue according to instruction AG. The execution of this instruction again increases the content of memory a or $\pi/2$. In the event of a positive result we continue according to instruction AL (for calculating w and s at m = 3). In the event of a negative result we continue according to instruction AH. This increases for the last time the content of memory a or $\pi/2$, after which the calculation for m = 4 is obtained according to instruction AM.

Instructions AI, AJ, AK, AL and AM are carried out for preparation of the values of w and s needed for the further computing for memories a and c according to the Table in the "Analysis" column. At m = 1, the execution of instruction AJ furnishes s = 1 in memory c. In the next instruction AJ, from $\pi/2$ set in memory b is subtracted the value v_1 set in memory a, and the result w is put back into memory a. After execution of these instructions, the machine continues according to instruction BA, which begins the actual calculation of the exponential series. At m = 2, the execution of instruction AK furnishes s = -1 in memory c, which is immediately followed by further calculation according to BA, since w = v_2 is already inserted into memory a. At m = 3, execution of AL furnishes s = -1 in memory c, which is immediately followed by instruction AJ, and the device continues according to BA. At m = 4, the instruction AM acts on memory c, and the operation proceeds further according to instruction BA.

Analysis	Vocabulary	Instruction: Index	Instruction: Operational Symbol	+	−	Remarks
	$0 = \langle 0\rangle$	IQ0	$0 \to \langle 0\rangle$			
	$1 = \langle 1\rangle$	IQ1	$1 \to \langle 1\rangle$			
x	$x = \langle a\rangle$	IQ2	$x \to \langle a\rangle$			
$y = \cos x$						
$\cos\lvert x\rvert = \cos x$						
$y = \cos\lvert x\rvert$	$\lvert x\rvert = \langle a\rangle$	AA	$\langle 0\rangle + \lvert\langle a\rangle\rvert \to \langle a\rangle$	AB	AB	
$\lvert x\rvert = z_0$	$\pi = \langle p1\rangle$	IQ3	$\pi \to \langle p1\rangle$			
$z_{n+1} = z_n - 2\pi \quad n = 1, 2, \ldots, N$	$2 = \langle p2\rangle$	IQ4	$2 \to \langle p2\rangle$			
$-2\pi \leq z_N < 0$	$2\pi = \langle b\rangle$	AB	$\langle p1\rangle . \langle p2\rangle \to \langle b\rangle$	AC	AC	
	$z_n = \langle a\rangle$	AC	$-\langle b\rangle + \langle a\rangle \to \langle a\rangle$	AC	AD	
	$z_N = \langle a\rangle$					
$z_N + \frac{m\pi}{2} = v_m, \quad m = 1, 2, 3, 4$	$\pi/2 = \langle b\rangle$	AD	$\langle p1\rangle : \langle p2\rangle \to \langle b\rangle$	AE	AE	
$0 \leq r_m < \pi/2$	$r_m = \langle a\rangle$	AE	$\langle a\rangle + \langle b\rangle \to \langle a\rangle$	AI	AF	
		AF	$\langle a\rangle + \langle b\rangle \to \langle a\rangle$	AK	AG	
$y = s . \cos w$		AG	$\langle a\rangle + \langle b\rangle \to \langle a\rangle$	AL	AH	
		AH	$\langle a\rangle + \langle b\rangle \to \langle a\rangle$	AM	AM	
	$s = \langle c\rangle$	AI	$\langle 1\rangle + \langle 0\rangle \to \langle c\rangle$	AJ	AJ	
	$w = \langle a\rangle$	AJ	$-\langle a\rangle + \langle b\rangle \to \langle a\rangle$	BA	BA	
		AK	$-\langle 1\rangle + \langle 0\rangle \to \langle c\rangle$	BA	BA	
		AL	$-\langle 1\rangle + \langle 0\rangle \to \langle c\rangle$	AJ	AJ	
		AM	$\langle 1\rangle + \langle 0\rangle \to \langle c\rangle$	BA	BA	

m	y	s	w
1	$\sin r_m$	1	$\frac{1}{2}\pi - r_m$
2	$-\cos r_m$	-1	r_m
3	$-\sin r_m$	-1	$\frac{1}{2}\pi - r_m$
4	$\cos r_m$	1	r_m

Fig.3.3 - First Part of Instructional Network in General Form

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

Analysis	Vocabulary	Instruction: Index	Instruction: Operational Symbol	Instruction: +	Instruction: -	Remarks
$\cos w \doteq S_j = 1 - \frac{w^2}{2!} + \frac{w^4}{4!} \cdots$	$w^2 = \langle a\rangle$	BA	$\langle a\rangle \cdot \langle a\rangle \to \langle a\rangle$	BB	BB	
$\cdots = \sum_{j=0}^{J} u_j$	$u_j = \langle b\rangle$	BB	$\langle 1\rangle + \langle 0\rangle \to \langle b\rangle$	BC	BC	
	$S_j = \langle d\rangle$	BC	$\langle 1\rangle + \langle 0\rangle \to \langle d\rangle$	BD	BD	
	$j = \langle e\rangle$	BD	$\langle 0\rangle + \langle 0\rangle \to \langle e\rangle$	BE	BE	
	$2j = \langle f\rangle$	BE	$\langle p2\rangle \cdot \langle e\rangle \to \langle f\rangle$	BF	BF	
$u_0 = S_0 = 1$	$(2j+1) = \langle f\rangle$	BF	$\langle f\rangle + \langle 1\rangle \to \langle f\rangle$	BG	BG	
	$(2j+2) = \langle g\rangle$	BG	$\langle f\rangle + \langle 1\rangle \to \langle g\rangle$	BH	BH	
	$(\ )\cdot(\ ) = \langle g\rangle$	BH	$\langle f\rangle \cdot \langle g\rangle \to \langle g\rangle$	BI	BI	
$u_{j+1} = -u_j \frac{w^2}{(2j+1)(2j+2)}$	$w^2/(\ )\cdot(\ ) = \langle g\rangle$	BI	$\langle a\rangle : \langle g\rangle \to \langle g\rangle$	BJ	BJ	
	$u_{j+1} = \langle b\rangle$	BJ	$-\langle b\rangle \cdot \langle g\rangle \to \langle b\rangle$	BK	BK	
	$\varepsilon = \langle p3\rangle$	IQ5	$\varepsilon \to \langle p3\rangle$			
$\lvert u_j\rvert - \varepsilon < 0$	$\lvert u_{j+1}\rvert - \varepsilon = \langle f\rangle$	BK	$-\langle p3\rangle + \lvert\langle b\rangle\rvert \to \langle f\rangle$	BL	CA	
$\lvert u_{j-1}\rvert - \varepsilon > 0$	$S_{j+1} = \langle d\rangle$	BL	$\langle b\rangle + \langle d\rangle \to \langle d\rangle$	BM	BM	
	$j+1 = \langle e\rangle$	BM	$\langle 1\rangle + \langle e\rangle \to \langle e\rangle$	BE	BE	$j+1 \to j$
$y = a \cdot S_j$	$y = \langle b\rangle$	CA	$\langle d\rangle \cdot \langle c\rangle \to \langle b\rangle$	CB	CB	
		CB	STOP			

Fig.3.4 - Second Part of Instructional Network in General Form

Computing the sum of the exponential series begins with instruction BA, which effects the replacement of the content of memory a with the value w^{α}. The next instructions BB, BC and BD produce the setting of the initial values u_j, s_j and j for $j = 0$ in memories b, d and e, where the values are replaced. Instructions BE, BF, BG, BH, BI and BJ carry out the calculation of the further members of the series according to the formula given in the "Analysis" column. The form also contains the command !Q5 for setting the constant ε in memory p3. This constant is used by the next instruction BK for verifying whether the iteration must be continued (according to instruction BL) or whether the machine is ready to proceed further with the computation (for which the instruction CA is valid). In instruction BL, the addition of the terms of series is carried out until the sum is reached. Instruction BM increases the content of memory e by unity, after which the machine proceeds with further iteration, beginning with instruction BE. Instruction CA effects the replacement of the content of b with the calculated y, after which instruction CB intervenes, which stops the computation.

The detailed instructional network is prepared on the forms of Figs.3.5, 3.6 and 3.7. On the first page of the "Model 2" form of Fig.3.5, in the first two lines, the memories 0000 and 0001 are entered, depicting the constants according to the commands !Q0 and !Q1. On the further lines of this page and nearly all of the second page (Fig.3.6) contain the instructions, each of them always in two lines. The third page of the form (Fig.3.7) gives the contents of the constant and variable memories in the course of the calculation. In the "Remarks" column there is always given on the corresponding line the coordination of the memory with the general address of the instructional network from the "Model 1" form.

The instructions are written in the manner described in Chapter 2. For example, the instruction on addresses 0002 and 0003 reads: "Take absolute value of number set in memory 0100, add to it the content of memory 0000, and set result in memory 0100 (replacing previous content); proceed further according to instructions

beginning with memory 0004".

Case: y = cos x

Model 2 Page 1

Index	Address		Entrance Information		Variable Information	Remarks
:Q0	0000	P	.00000000 0 00 0	B	0	
:Q1	0001		.40000000 0 01 0	B	1	
AA	0002		0100 0004 SJ			
	0003	P	0100 0004 0000			
AB	0004		0101 0006 N			
	0005	P	0110 0006 0111			
AC	0006	P	0100 0008 SM			
	0007		0100 0010 0101			
AD	0010		0101 0012 D			
	0011	P	0111 0012 0110			
AE	0012	P	0100 0022 S			
	0013		0101 0014 0100			
AF	0014	P	0100 0026 S			
	0015		0101 0016 0100			
AG	0016		0100 0030 S			
	0017	P	0101 0020 0100			
AH	0020		0100 0032 S			
	0021	P	0101 0032 0100			
AI	0022	P	0102 0024 S			
	0023		0000 0024 0001			
AJ	0024	P	0100 0034 SM			
	0025		0101 0034 0100			
AK	0026		0102 0034 SM			
	0027	P	0000 0034 0001			
AL	0030	P	0102 0024 SM			
	0031		0000 0024 0001			
AM	0032		0102 0034 S			
	0033	P	0000 0034 0001			
BA	0034		0100 0036 N			
	0035	P	0100 0036 0100			
BB	0036	P	0101 0040 S			
	0037		0000 0040 0001			

Fig.3.5 - First Part of Instructional Network

STAT

Case: $y = \cos x$

Model 2

Index	Address		Entrance Information	Variable Information	Remarks
BC	0040		0103 0042 S		
	0041	P	0000 0042 0001		
BD	0042	P	0101 0044 S		
	0043		0000 0044 0000		
BE	0044	P	0105 0046 A		
	0045		0104 0040 0111		
BF	0046		0105 0050 S		
	0047	P	0001 0050 0105		
BG	0050	P	0106 0052 S		
	0051		0001 0052 0105		
BH	0052		0106 0054 A		
	0053	P	0106 0054 0105		
BI	0054		0106 0056 D		
	0055	P	0106 0056 0100		
BJ	0056	P	0101 0060 AM		
	0057		0106 0060 0101		
BK	0060	P	0105 0062 SMA		
	0061		0101 0066 0112		
BL	0062		0103 0064 S		
	0063	P	0103 0064 0101		
BM	0064		0104 0044 S		
	0065	P	0104 0044 0001		
CA	0066	P	0101 0070 A		
	0067		0102 0070 0103		
CB	0070		0000 0000		
	0071		0000 0000 0000		
	0072	P			
	0073				
	0074	P			
	0075				
	0076				
	0077	P			

Fig.3.6 – Second Part of Instructional Network

STAT

Case: y = cos x

Model 2 Page 3

Index	Address	Entrance Information	Variable Information	Remarks
Q2	0100		B $x, \lvert x \rvert, z_m, z_y, r_m, w, w_s$	a 100
	0101 P		$2\pi, \pi/2, u_j, u_{j+1}, y$	b 101
	0102 P			c 102
	0103		S_j, S_{j+1}	d 103
	0104 P		$j, j+1$	e 104
	0105		$2j, 2j+1, u_{j+1}, u_{j+1}$	f 105
	0106		$2j+2, (\ ) \cdot (\ ), w^2(\ ) \cdot (\ )$	g 106
	0107 P			
Q3	0110 P	.02207733 0 02 0	B π	p1 110
Q4	0111	.10000000 0 02 0	B 2	p2 111
Q5	0112	.10000000 1 23 0	B 2^{29}	p3 112
	0113 P			
	0114			
	0115 P			
	0116 P			
	0117			
	0120 P			
	0121			
	0122			
	0123 P			
	0124			
	0125 P			
	0126 P			
	0127			
	0130			
	0131 P			
	0132 P			
	0133			
	0134 P			
	0135			
	0136			
	0137 P			

Fig.3.7 – Third Part of Instructional Network

CHAPTER 4

INVESTIGATION OF A CENTERED OPTICAL SYSTEM WITH THE AUTOMATIC CALCULATOR

We conclude the first part of this Manual with two concrete examples of the application of the automatic computer. The first of the examples is from the field of geometrical optics.

INTRODUCTION

In calculating a high-quality optical-system, the most important part of the working procedure is the elimination of the imaging error with the help of the formative parameters (dimensions). This must be done by calculating the ray path of the given optical system and that of the optical system obtained from this system by changing one of the formative parameters by a small amount. For example, in a photographic objective the formative parameters are usually more than twenty. This means that we must carry out besides the fundamental calculation of the ray path at least twenty other similar calculations. It is not sufficient here to calculate the path of one ray. At least five rays must be calculated for each point of the image, and three points of the image already require the calculation of 300 ray paths.

Of course, by formulary calculation with the help of Tables and the computer, work and time are saved. This suggested the idea of carrying out the variation of the formative parameters solely on the basis of the ray path contained in the plane of symmetry of the optical system, because its calculation is relatively simple. The calculation of a ray path convergent with the optical axis is not mathematically difficult, but time-consuming. With the help of the computer, however, it is very easy to prepare the instructional network, according to which the device calculates in a short time a ray path convergent with the optical axis.

The construction of the instructional networks for the calculation of ray paths

divergent and convergent with the optical axis will be explained. It should be noted that in the preparation we proceeded more on the basis of decadic considerations than the economy of the instructional networks. However, for the present capacity of the computer, there is sufficient room for setting instructions for the calculation of the sum of the squares of errors and instructions for minimizing these. The discussion of such a supplementary working procedure is beyond the scope of the present work.

STAT

Geometric Analysis of the Problem

4.1. Formulation

The investigation of a centered optical system involves a repetition of the solution for the problem of the transition of a given ray through the given optical system. This problem can be formulated roughly as follows:

Assume a system with a certain number m of spherical areas with their centers in the axis x_1. The last of these areas is the focal plane. The remaining part of the areas have the form of the surface of the individual lenses of the optical system. This part of the areas will be called the boundary. Figure 4.1 shows examples of convex, concave, and plane boundaries of a focal plane. Each of the given m areas has a certain constant N_j (relative refractive index, i.e., the ratio of the refractive indexes in front and behind the boundary*). The ray is further identified by its individual directional vector and point. The intersection of the given ray with the first boundary must be determined. In addition, the new individual vector giving the direction of the ray after refraction must be determined (with the help of the constant N_1. This refracted ray is then again determined by the individual directional vector and the point (intersection with the first boundary). Then the intersection of the refracted ray with the second boundary must be determined, and then the new direction of the ray, and so forth, until the focal plane is reached. In this case only the intersection of the ray with this plane, must be determined and the problem of the transit of the given ray in the optical system is solved.

This shows that with the solution the problem is practically reduced to a few repetitions of the solution of two such geometrical problems**.

---

* N_j is a constant as long as only the wavelength of the light is considered.

** The points will be denoted the same as their half vector, so that we may speak of point $x_j \equiv (x_{j,1}, x_{j,2}, x_{j,3})$. The variable point (vector) will be denoted $x \equiv (x_1, x_2, x_3)$.

[redacted]

STAT

1. Determination of the Intersection

Given a straight line with its individual directional vector $a_{j-1} \equiv (a_{j-1,1}, a_{j-1,2}, a_{j-1,3})$ and point $x_{j-1} \equiv (x_{j-1,1}, x_{j-1,2}, x_{j-1,3})$. Determine the intersection $x_j \equiv (x_{j,1}, x_{j,2}, x_{j,3})$ of this straight line with the spherical area given by the center $s_j \equiv (S_{j,1}, 0, 0)$ and the radius R_j (or with the plane $x_1 = S_{j,1}$).

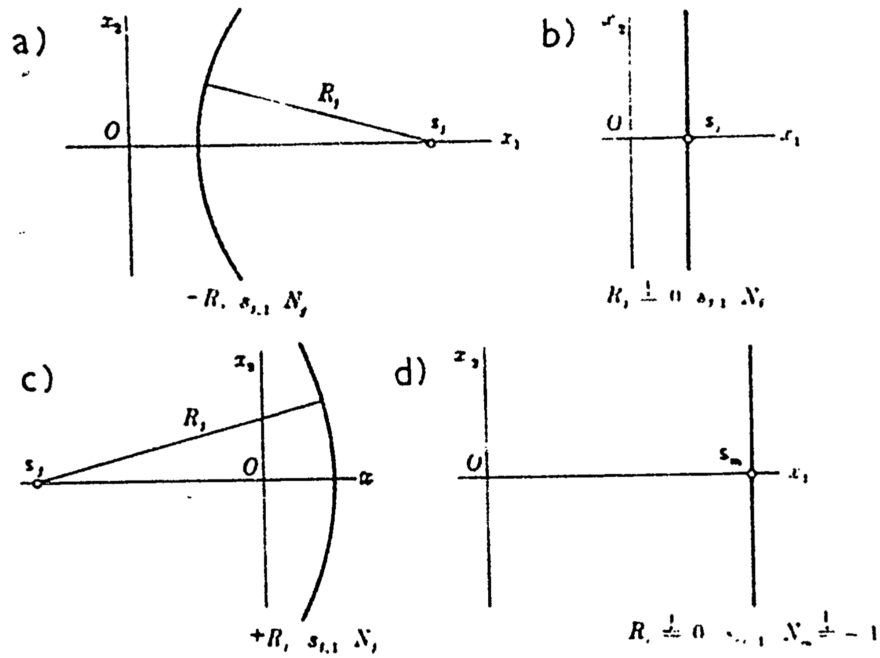


Fig.4.1 - Examples for Types of Areas in an Optical System

a) Convex boundary; b) Plane boundary; c) Concave boundary; d) Focal plane

2. Determination of the New Direction

a) Determine the uniform vector $n_j \equiv (n_{j,1}, n_{j,2}, n_{j,3})$ normal to the area (given in problem 1) in the point x_j.

b) Determine the uniform vector $a_j \equiv (a_{j,1}, a_{j,2}, a_{j,3})$, which is the linear combination of the vectors a_{j-1} and n_j (i.e., located in the same plane; see Fig.4.2) and for whose angle β_j with the vector n_j there is valid $\sin \beta : \sin \alpha_j = N_j$; N_j is

136

given by the constant and a_j if the angle of the vectors n_j and a_{j-1}.

At the solution of this geometrical problem, some facts must be taken into account according to the physical sense of the problem. These facts are considered in detail in paragraphs 4.6–4.11.

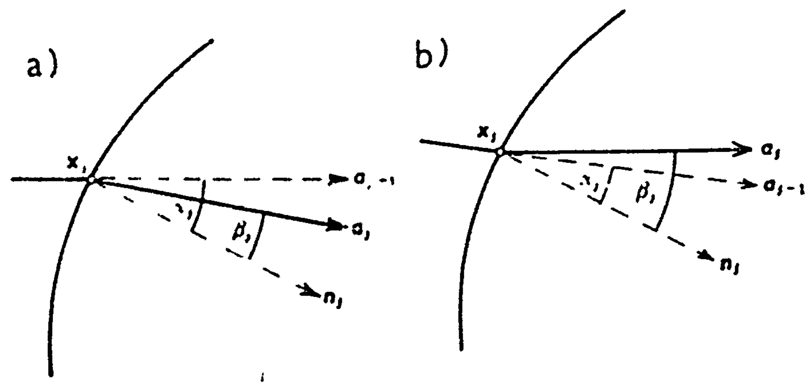


Fig.4.2 - Transit of j Rays at Boundary

a) For $N_j < 1$, b) For $N_j > 1$

4.2. Transition of a Ray Through a Spherical Boundary

We describe the procedure of the calculation for the stage where the transition of a j ray is investigated in an area. It should first be noted that this area is spherical. Known in this case are:

the straight lines

$$\mathbf{x} = \mathbf{x}_{j-1} + \mathbf{a}_{j-1}t, \qquad |\mathbf{a}_{j-1}| = 1$$

$$\begin{aligned} x_1 &= x_{j-1,1} + a_{j-1,1}t \\ x_2 &= x_{j-1,2} + a_{j-1,2}t \\ x_3 &= x_{j-1,3} + a_{j-1,3}t . \end{aligned}$$

the spherical area

$$(\mathbf{x} - \mathbf{s}_j)^2 = R_j^2, \quad \text{t. j.} \quad (x_1 - s_{j,1})^2 + x_2^2 + x_3^2 - R_j^2 = 0 .$$

the constant N_j.

STAT

1. Calculate the Intersection of the Ray with the Spherical Surface

a) Determine the value of the parameter t for the intersection:

$$(x_{j-1,1} + a_{j-1,1}t - s_{j,1})^2 + (x_{j-1,2} + a_{j-1,2}t)^2 + (x_{j-1,3} + a_{j-1,3}t)^2 = R_j^2 = 0$$

$$\underbrace{(a_{j-1,1}^2 + a_{j-1,2}^2 + a_{j-1,3}^2)}_{1}t^2 + 2\underbrace{(a_{j-1,1}x_{j-1,1} + a_{j-1,2}x_{j-1,2} + a_{j-1,3}x_{j-1,3} - a_{j-1,1}s_{j,1})}_{A_j}t$$

$$+ \underbrace{(x_{j-1,1}^2 + x_{j-1,2}^2 + x_{j-1,3}^2 - 2x_{j-1,1}s_{j,1} + s_{j,1}^2 - R_j^2)}_{B_j} = 0$$

$$t^2 + 2A_jt + B_j = 0,$$

$$t_j = -A_j \pm \sqrt{A_j^2 - B_j}\ ^{*)} \tag{4.1}$$

b) Calculate the coordinates of the intersections:

$$x_{j,1} = x_{j-1,1} + a_{j-1,1}t_j, \quad x_{j,2} = x_{j-1,2} + a_{j-1,2}t_j, \quad x_{j,3} = x_{j-1,3} + a_{j-1,3}t_j. \tag{4.2}$$

2. Calculate the Vector a_j

Since a_j must be the linear combination of the vectors a_{j-1} and n_j, it will have the form of $a_j = C_ja_{j-1} + K_jn_j$ (C_j and K_j are constants which must be determined).

a) Determine the vector $n_j \equiv (n_{j,1}, n_{j,2}, n_{j,3})$: The vector joining the point x_j of the spherical area with its center s_j has its direction normal to point x_j and magnitude R_j. Combine the uniform vectors normal to n_j, then

$$n_{j,1} = (x_{j,1} - s_{j,1}) : R_j, \quad n_{j,2} = x_{j,2} : R_j, \quad n_{j,3} = x_{j,3} : R_j\ ^{*)} \ ** \tag{4.3}$$

b) Calculate the constants C_j, K_j and combine the vectors a_j. The sought vectors a_j must have the following three characteristics:

$$a_j = C_ja_{j-1} + K_jn_j, \tag{4.4}$$

$$|a_j| = 1, \tag{4.5}$$

$$\sin \beta_j = N_j \sin \gamma_j, \tag{4.6}$$

---

* We select "+" for concave boundaries and "-" for convex (See paragraphs 4.9 and 4.13.

**Reorientation to the normal (See paragraphs 4.7 and 4.13.

STAT

199

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

where α_j is the angle of the vectors a_{j-1}, n_j, and β_j is the angle of the vectors a_j, n_j.

We multiply eq.(4.4) by the scalar vector n_j and obtain

$$a_j \cdot n_j = C_j a_{j-1} \cdot n_j + K_j n_j \cdot n_j . \tag{4.7}$$

Since a_j, a_{j-1}, n_j are uniform vectors, the following is valid:

$$a_j \cdot n_j = \cos\beta_j , \quad a_{j-1} \cdot n_j = \cos\alpha_j , \quad n_j \cdot n_j = 1 ,$$

so that instead of (4.7) we write

$$\cos\beta_j = C_j \cos\alpha_j + K_j ,$$

and then

$$K_j = \cos\beta_j - C_j \cos\alpha_j . \tag{4.8}$$

We now multiply eq.(4.4) by the scalar vector a_{j-1} and obtain

$$a_j \cdot a_{j-1} = C_j a_{j-1} \cdot a_{j-1} + K_j n_j \cdot a_{j-1} ,$$

i.e.,

$$\cos(\beta_j - \alpha_j) = C_j + K_j \cos\alpha_j .$$

We separate the left side and get to K_j according to eq.(4.8):

$$\cos\beta_j \cos\alpha_j + \sin\beta_j \sin\alpha_j = C_j + (\cos\beta_j - C_j\cos\alpha_j)\cos\alpha_j$$
$$\cos\beta_j \cos\alpha_j + \sin\beta_j \sin\alpha_j = C_j(1 - \cos^2\alpha_j) + \cos\alpha_j \cdot \cos\beta_j$$
$$\sin\beta_j \sin\alpha_j = C_j \sin^2\alpha_j$$
$$C_j = \frac{\sin\beta_j}{\sin\alpha_j} = N_j . \tag{4.9}$$

Proceeding from C_j to (4.8), we obtain

$$K_j = \cos\beta_j - N_j \cos\alpha_j .$$

For $\cos\beta_j$, according to (4.6), we have

$$\cos^2\beta_j = 1 - \sin^2\beta_j = 1 - N_j^2 \sin^2\alpha_j = 1 - N_j^2(1 - \cos^2\alpha_j) ,$$

so that

$$\cos\beta_j = + \sqrt{1 - N_j^2(1 - \cos^2\alpha_j)} \;* \tag{4.10}$$

$$K_j = + \sqrt{1 - N_j^2(1 - \cos^2\alpha_j)} - N_j \cos\alpha_j . \tag{4.11}$$

* The selection of "+" for the root follows from paragraph 4.7. If the root comes out imaginary, this corresponds to total reflex; see paragraph 4.11.

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

Since a_{j-1}, n_j are known, we easily find the value $\cos x_j = a_{j-1} \cdot n_j$, and hence also the value K_j.

Calculation of the factors

$$a_{j,1} = C_j a_{j-1,1} + K_j n_{j,1},\ a_{j,2} = C_j a_{j-1,2} + K_j n_{j,2},\ a_{j,3} = C_j a_{j-1,3} + K_j n_{j,3}$$

completes the calculation of the transition of j rays through (spherical) areas.

4.3. Transition of Rays at a Plane Boundary

If the j area is a plane of equation $x_1 = s_{j,1}$, it is simpler to calculate the intersection of the ray with this plane. Part 1a) is calculated and replaces part 1a'):

$$x_{j-1,1} + a_{j-1,1} t = s_{j,1}.$$

We assume

$$a_{j-1,1} \neq 0,^{*)} *$$

so that

$$t_j = (s_{j,1} - x_{j-1,1}) : a_{j-1,1}. \qquad (4.12)$$

The further calculation is the same as that for the spherical area, but part 2a is omitted because the vector normal to the considered plane is known: $n_j \equiv (1, 0, 0)$.

So far it has been assumed that the transit of one (given) ray in the optical system is being investigated. It is, however, necessary to select a suitably resolved ray with various directions and various places of origin in the optical system.

4.4. Selection of the Place of Origin

The plane (x_2, x_3) is selected in such a way that all of the points belonging to the part of the area coming into consideration for the first boundary have a positive but minimal coordinate x_1. If this first boundary is convex or plane, the beginning of the coordinates is located in the intersection of the optical axis of the system with this boundary. In the plane (x_2, x_3) a lattice is given of 28 points

* Concerning exclusion of the case $a_{j-1,1} = 0$ see paragraph 4.6.

STAT

according to Fig.4.3, where Δ is a given (by definition) length. If a ray arising in the optical system in a given direction is to be investigated, we select in the given direction each of the points of this lattice with the starting point in this direction, and express it by $x_0 \equiv (x_{01}, x_{0,2}, x_{0,3})$. The order in which we select the points of the lattice is indicated in Fig.4.3 by arrows.

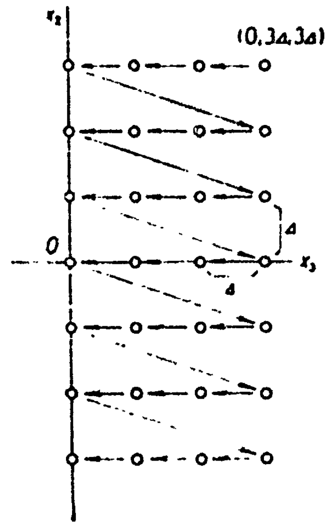


Fig.4.3 - Selection of the Place of Origin of a Ray in the Optical System

4.5. Selector of Direction

The direction of the entering ray is determined by the connection between the points $p \equiv (p_1, 0, 0)$ and $q \equiv (0, q_2, 0)$ oriented from q to p (see Fig.4.4). This $p_1 > 0$ is constant (selected in dependence on the angle of vision of the given optical system), and with q_2 we select in succession the points 5, 4, 3, 2, 1, 0. For the uniform vector $a_0 \equiv (a_{0.1}, a_{0.2}, a_{0.3})$ in this particular direction, is valid:

$$a_{0,1} = p_1 : \sqrt{p_1^2 + q_2^2},$$
$$a_{0,2} = -q_2 : \sqrt{p_1^2 + q_2^2},$$
$$a_{0,3} = 0. \quad (4,13)$$

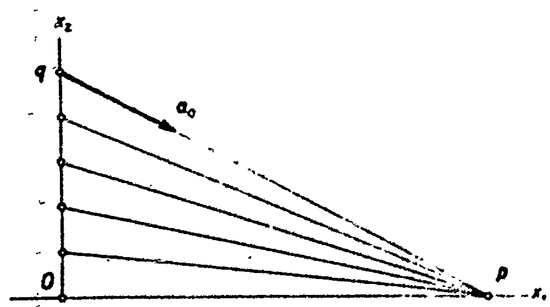


Fig.4.4 - Selection of Entering Direction of Ray

We calculate then for the given optical system the paths of the rays in six different directions and for each of these directions we investigate 28 different places of origin of the rays in the optical system, and hence we calculate in all 168 rays.

STAT

according to Fig. 4.3, where Δ is a given (by definition) length. If a ray arising in the optical system in a given direction is to be investigated, we select in the given direction each of the points of this lattice with the starting point in this direction, and express it by $x_0 \equiv (x_{01}, x_{0,2}, x_{0,3})$. The order in which we select the points of the lattice is indicated in Fig. 4.3 by arrows.

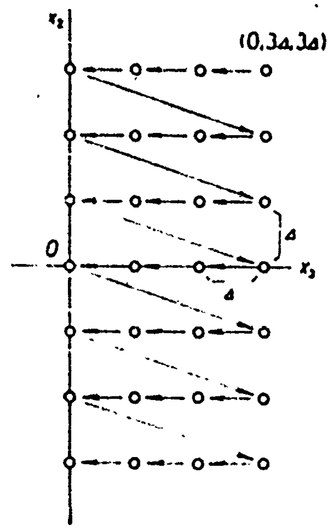


Fig. 4.3 - Selection of the Place of Origin of a Ray in the Optical System

4.5. Selector of Direction

The direction of the entering ray is determined by the connection between the points $p \equiv (p_1, 0, 0)$ and $q \equiv (0, q_2, 0)$ oriented from q to p (see Fig. 4.4). This $p_1 > 0$ is constant (selected in dependence on the angle of vision of the given optical system), and with q_2 we select in succession the points 5, 4, 3, 2, 1, 0. For the uniform vector $a_0 \equiv (a_{0.1}, a_{0.2}, a_{0.3})$ in this particular direction, is valid:

$$a_{0,1} = p_1 : \sqrt{p_1^2 + q_2^2},$$
$$a_{0,2} = -q_2 : \sqrt{p_1^2 + q_2^2},$$
$$a_{0,3} = 0. \quad (4,13)$$

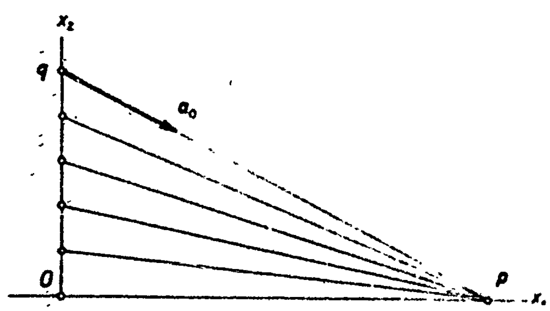


Fig. 4.4 - Selection of Entering Direction of Ray

We calculate then for the given optical system the paths of the rays in six different directions and for each of these directions we investigate 28 different places of origin of the rays in the optical system, and hence we calculate in all 168 rays.

349

4.6. Impermissible Angle of Ray with Optical Axis

If the positive direction of the ray includes the positive direction of the axis x_1, the ray loses significance for our purpose. If, therefore, the directional cos $a_{j,1}$ satisfies the inequality

$$a_{j,1} < \varepsilon \tag{4,14}$$

at $\varepsilon > 0$, by definition we cease to investigate the path of such a ray (here, among others, the case $a_{j-1,1} = 0$ in eq.4.12 is included).

4.7. Orientation of a Normal Vector

With a convex boundary we orient the normal vector in the direction toward the center of the spherical area and with a concave boundary in the direction away from the center. This assures that the angle β of the directional vector of the ray with the normal vector, in the considered case is always acute, so that cos $\beta_j < 0$ is excluded (which justifies selection of the "+" sign with the root in (4.10)).

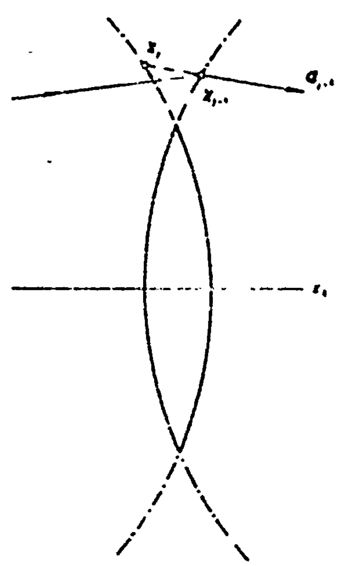


Fig.4.5 - Transit of a Ray through an Impermissible Part of the Area

4.8. Imaginary Intersection

Significant for our problem is only the real intersection of the ray with the concerned area. If, therefore, the ray does not encounter a real spherical area, we do not calculate the ray and we cease to investigate the path of this ray (in some calculations, according to eq.(4.1), we therefore, exclude the case $A_j^2 - B_j < 0$).

4.9. Selection of One or Two Real Intersections

For a convex boundary, of two real intersections of the ray with the concerned

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

spherical area, only the one can have significance which has the smaller coordinate x_1. Therefore the other intersection is not calculated. Analogously, for a concave boundary we calculated only the intersection with the greater coordinate x_1 (accordingly, the sign of the root is selected for some of the calculations according to eq.(4.1).

4.10. Impermissible Part of Area

Figure 4.5 shows the significance of a part of an area which can never be used in the given optical system. If the ray (as indicated in Fig.4.5) intersects any part of the area in this portion, it loses significance for the problem. We cease to investigate its path. In calculating this case, it is found that the parameter t_j of the intersection of the ray with the additional area (i.e., the point x_j) comes out negative.

4.11. Total Reflex

It is necessary to exclude the case where total reflex occurs at any boundary. This case is found when, at the calculation the direction, the ray, after transit through the considered boundary, expressed below the root in eq.(4.10), comes out negative or zero. Such a case is not further investigated.

4.12. Distinguishing between Spherical and Plane Boundaries

In the first part of the calculation of the transit of the ray through the boundary it must be decided whether the considered area is spherical or plane. In order to enable the machine easily to specify the proper calculation, we assign to every plane the value $R_j = 0$ (although this does not have geometrical sense!), while for spherical areas we always have the radius $R_j \neq 0$.

4.13. Distinguishing between Convex and Concave Boundaries

It has been found advantageous to assign to the radius R_j with convex boundaries the "-" sign and with the concave boundaries the "+" sign, and then to set the

STAT

values of the radii in the machine with those signs. In this way the machine is able in advance, at the calculation according to eq.(4.1) to instruct where the "+" or the "-" sign for the root is to be selected; it is sufficient to give the root the same sign as R_j has. Further, such assignment of the sign to the radius R_j (in calculating according to (4.3)) is assured in agreement with the normal vector (see Paragraph 4.7). Starting here, the symbol R_j is defined as the radius already provided with the corresponding sign.

4.14. Distinguishing between Focal Plane and Boundary

We require an element with whose help the machine is instructed to change the procedure in the moment when the calculation of the path of the ray is finished, i.e., when the ray reaches the focal plane. An element is used with which the constant $N_m = -1$ is assigned to the focal plane (although this does not have physical sense[1]). For all of the remaining areas we thus have $N_j > 0$.

4.15. Characterization of Boundary Constants

On the basis of what is stated in the three preceding paragraphs, the j boundary is fully characterized by the constants R_j, $s_{j,1}$, N_j. These constants indicate the size and location of the boundaries in the system, as well as those of the other considered areas. This is utilized for the construction of the instructional network.

DESCRIPTION OF CONSTRUCTION OF INSTRUCTIONAL NETWORK

4.16. Group Arrangement

The instructional network is diagrammatically represented in the group arrangement of Fig.4.6. For the detailed investigation of the network it is advantageous to use the "model 1" forms shown in Figs.4.7-4.13. A survey of the storage of the necessary constants in the memory of the machine will be found in the three pages of the "model 2" form shown in Figs.4.21-4.23.

STAT

Until the instructions in some part of the instructional network are established in a particular sequence, no mention is made of individual instructions. The entire group of such instructions is denoted in Fig.4.6 by a rectangle. Some groups will include the part of the instructional network in which the instructions are established by a somewhat more complicated procedure. Thus, for example, the group of instructions FA to FZ (briefly, the group F) contains, besides others, the instructions FQ to FU, according to which the machine repeats the root-extracting cycle several times. The cycle is described in detail in paragraph 1.5. The group F in Fig.4.6 is denoted by one rectangle.

The manner in which the individual groups can be established is indicated by arrows in the grouped scheme.

For the solution of the given problem, it is necessary to be able to solve other problems. This article will explain the parts of the instructional network with whose help the further problems are solved in the machine.

4.17. <u>Problem 1.</u>

Given is the boundary with its constants R_j, $s_{j,1}$, N_j (the constants are set in accordance with the remarks presented in paragraphs 4.12-4.15). Further given is the ray falling on the boundary with its constants $a_{j-1,1}$, $a_{j-1.2}$, $a_{j-1.3}$, $x_{j-1.1}$, $x_{j-1.2}$, $x_{j-1.3}$.

To be calculated are the constants of the same ray after refraction at the boundary.

Problem I is solved with the part of the instructional network for instruction EA to instruction NZ. Instruction JA and the address relating to instruction VA are in this case disregarded. The given constants are stored in the memories n0, n1, ..., n10 (concretely 460, ..., 470)*.

The constants a_{j1}, $a_{j,2}$, $a_{j,3}$, $x_{j,1}$, $x_{j,2}$, $x_{j,3}$ replace in the respective

* c.f. paragraph 4.22

STAT

639

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

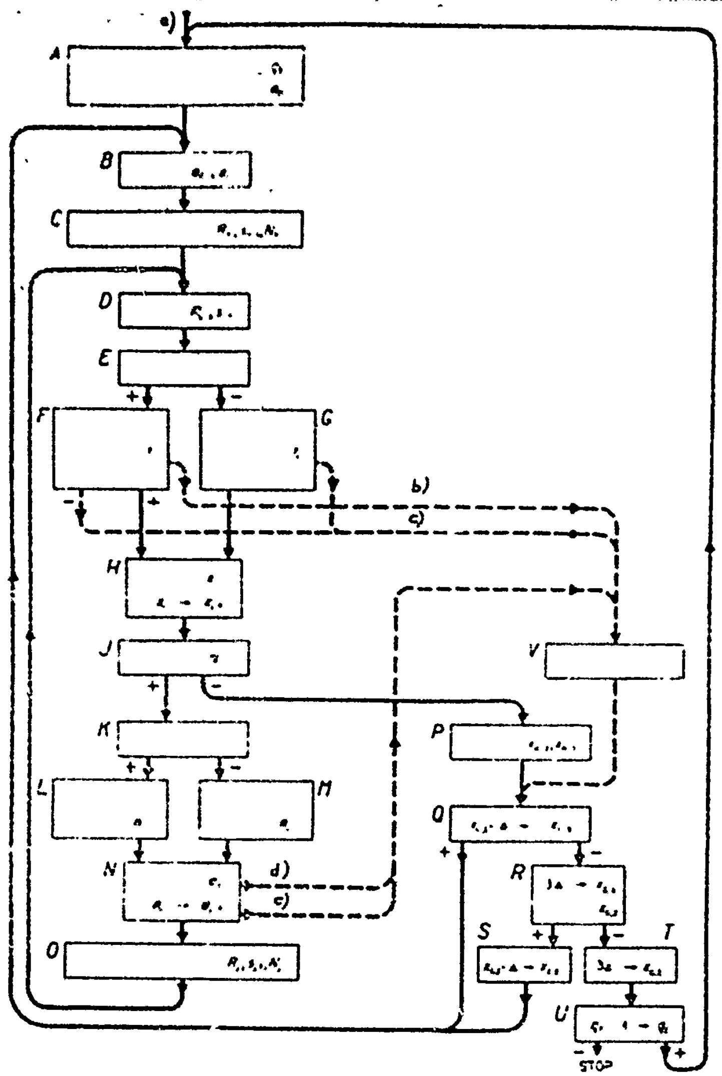


Fig. 4.6 - Group Scheme

a - Start (A) Exit of initial direction q_2 Determination of components of vector a_0, x_0 (C) Setting of addresses for R_j, s_j, 1, N_j (D) Storage of R_j, $S_{j,1}$ (E) Sphere-plane (F) Calculation of parameter t_j of intersection (G) Calculation of parameter t_j of intersection b - Imaginary intersection c - Impermissible part (H) Calculation $x \rightarrow x_{j-1}$ (J) Storage N_j (K) Sphere-plane (L) Calculation normal n_j (M) Calculation normal n_j (N) Calculation a_j, $a_j \rightarrow a_{j-1}$ d - Total reflex; e - Great angle (O) Increasing of addresses for R_j, $s_{j,1}$, n_j (V) Empty exit; (P) Exit $x_{m,2}$, $x_{m,3}$ (R) Trial $x_{0,2}$ $3\Delta \rightarrow x_{0,3}$ (S) $x_{0,2} - \Delta \rightarrow x_{0,2}$ (T) $3\Delta \rightarrow x_{0,2}$ (U) $q_2 - 1 \rightarrow q_2$

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

memories the corresponding constants $a_{j-1.1}$, $a_{j-1.2}$, $a_{j-1.3}$.

For the purpose of instruction EA or KA an investigation is made whether the considered boundary is a spherical or a plane area. According to this instruction the expression

$$|R_j| - \epsilon . \tag{4.15}$$

is calculated. The value ϵ is a positive constant given in such a way that it is smaller than the radius of any of the given spherical areas. For any spherical area, therefore, the expression (4.15) comes out positive, while for any plane area, where we always have $R_j = 0$ (c.f. paragraph 4.12), it comes out negative.

If the problem concerns a sphere, the group of instructions F is valid. These instructions are based on the first two constants of the boundaries and on all of the constants of the straight lines (stored in the respective memories). The calculated parameter is stored in memory h7 (concretely 427). The instructions of group F serve for calculating according to the procedure described in the first part of paragraph 4.2. If the problem concerns a plane, the instructional group G is valid. The latter is based on the same kind of calculating information as group F, and is intended for equivalent information, whose results are stored in the same memories as group F. The calculation is carried out in the manner described in paragraph 4.3 for planes.

The instructional group H, which is common to spheres and planes, starts with a known constant of the straight line and a known parameter of the intersection, stored in the respective memories. This value serves for calculating the coordinate of the intersections $x_{j.1}$, $x_{j.2}$, $x_{j.3}$. The resultant coordinate replaces the given constant $x_{j-1.1}$, $x_{j-1.2}$, $x_{j-1.3}$, which is no longer needed. The instructions of this group serve for the calculating procedure according to eq.(4.2).

The further calculating procedure in turn depends on the second area. It is determined by instruction KA. If the problem concerns a sphere, instructional group L is valid, which, with a coordinate of the points $x_{j.1}$, $x_{j.2}$, $x_{j.3}$ on the area and

STAT

a given constant of the area R_j, $s_{j.1}$, determines the components of the uniform vector to the normal, and is stored in memories h10, h11, h12 (concretely 430, 431, 432). Proceed according to eq.(4.3). If the problem concerns a plane, the uniform vector to the normal is known in advance. Instructional group M in this case only stores the known vectors (1, 0, 0) in the corresponding memories, which require a vector $n_j \equiv (n_{j.1}, n_{j.2}, n_{j.3})$.

The further procedure is common to spheres and planes. Instructional group N is employed. The latter is based on a known uniform directional vector a_{j-1} of the ray before refraction, a uniform vector to the normal n_j, and a known mutual index of refraction N_j. All of the values are already stored in the memories. In this procedure, the uniform directional vector $a_j \equiv (a_{j.1}, a_{j.2}, a_{j.3})$ of the ray after refraction is determined according to the second part of paragraph 4.2. There vectors replace the given directional vectors a_{j-1} in the corresponding memories.

This completes the description of the part of the instructional network solving problem I.

4.18. Problem II

Given is the entire optical system with its constants R_1, $s1.1$, N_1; R_2, $s2.1$, N_2; ...; R_m, $s_{m.1}$, N_m. Given also is the entering ray with its constants $a_{0.1}$, $a_{0.2}$, $a_{0.3} = 0$; $x_{0.1} = 0$, $x_{0.2}$, $x_{0.3}$.

To be calculated are the two coordinates $x_{m.2}$, $x_{m.3}$ of the points in the focal plane where the ray hits after passing through the optical system*.

Problem II is solved with the part of the instructional network BA to OC together with instructions PA and PB. The constants of the optical system are stored in memories p0, p1, p2, ... (concretely 500, 501, 502, ...). The constants of the entering ray $a_{0.1}$, $a_{0.2}$, $x_{0.2}$, $x_{0.3}$ are in memories k1, k2, m0, m1 (concretely 441,

* The first coordinate point in the focal plane is denoted $x_{m.1} = s_{m.1}$.

STAT

442, 450, 451)*.

The resultant values are perforated into the cards by the machine.

In order to be able to use the part of the instructional network solving problem I, it is necessary to be able to fill the required memories. We begin with the first boundary, so that in memories n0, n1, n2, (concretely 460, 461, 462) must be stored the constants R_1, $s_{1.1}$, N_1. These, together with the remaining constants of the system are stored in memories p0, p1, p2, ... (concretely 500, 501, 502, ...).

The instructions leading to transmission of the constants are:

DA	$\langle p0 \rangle - \langle 0 \rangle \cdot \langle n0 \rangle$.
DB	$\langle p1 \rangle - \langle 0 \rangle \cdot \langle n1 \rangle$.
JA	$\langle p2 \rangle - \langle 0 \rangle \cdot \langle n2 \rangle$.

Moreover, it is necessary to replace, at least, in memories n3 to n10 (concretely 463 to 470), the constants of the entering ray by the constants of the incidence of the ray at the boundary. These values are stored in memories k1, k2, m0, m1.

We therefore also give instructional group B by which this is accomplished.

According to these given instructions, the machine must begin by solving problem I for the first boundary. The constants of the ray after refraction, calculated as results, form the basis for the solution of problem I for the second boundary. The machine is provided directly with the corresponding memories. In order to be able to begin with the solution of problem I for the second boundary, also the constants R_2, $s_{2.1}$, N_2 must be stored in place of the constants of the first boundary.

This is accomplished with instructions DA, DB, JA, as explained above, except that the corresponding addresses are suitably changed. Since the contents of memories p0, p1, p2, p3, ... are full, it is necessary in each of the instructions to

---

* The values $a_{0.3}$, $x_{0.1}$ are not reserved for particular memories, because $a_{0.3} = x_{0.1} = 0$.

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

augment the respective address by three.

This change is accomplished by the instructions of group 0, of which we cite the first:

OA $\quad C!(DA+1 \to g10 \to !DA \cdot 1 \cdot^{10})$ *

This instruction as changed by instruction DA is as follows:

DA $\quad (p3) \to (0) \to (n0).$

Similarly, instructions OB and OC are transformed by instructions DB and JA.

After this, instructions DA, DB, and JA can be properly used for solving problem I for the second boundary, etc.

The process is repeated until the ray reaches the focal plane. Then the negative sign of the constant N_m (c.f. paragraph 4.14) produces the effect that the result of the operation according to instruction JA comes out negative (a negative number is regarded as zero). Then in second address of the further procedure, i.e., PA is valid. The machine does not further calculate the direction of the ray, but perforates the card with the coordinates of the ray, which are already the required results.

As can be seen, in the moment when the machine finishes problem II, the instructions DA, DB, JA are changed. If the machine returns later to the solution of problem II, these instructions must always be brought back to the original state. In every calculation of problem II this is done according to instructions CA, CB, CC.

This completes the description of the part of the instruction network solving problem II.

4.19. Problem III

Given is the whole optical system with its constants R_1, $s_{1.1}$, N_1; R_2, $s_{2.1}$,

---

* Instruction OA means: Start with the second word of instruction DA, to whose third part (i.e., to the address for R_1), add the content of memory g10 (i.e., the number 3), so that the changed word replaces the original word.

STAT

315

N_2; ...; R_m, $s_{m.1}$, N_m. Also given is the direction a_0 of the entering ray with the constants $a_{0.1}$, $a_{0.2}$, $a_{0.3} = 0$. To be solved is problem II for 28 points selected in succession after the starting point $x_0 \equiv (x_{0.1}, x_{0.2}, x_{0.3})$ according to the system described in paragraph 4.4.

Problem III is solved with the instructions that were necessary for solving problem II supplemented by the instructions QA, RA, RB, SA. All of the given values are stored in the same memories as in problem II. Instead of the coordinates of the starting point (i.e., in memories m0 and m1), the coordinates of this point of the lattice shown in Fig.4.3 are stored, with which the start has to be made, i.e., points (0, 3Δ, 3Δ).

This satisfies all of the conditions needed to begin the solution of problem II for this point.

After exit of the results (with instructions PA, PB), the solution of problem II must be repeated for the further points of the lattice. In view of the change in the position of the starting point, instruction QA is used, which changes its third coordinate by Δ. Then it is possible to return to instruction BA. If in any problem II $x_{0.3} = 0$, the result of operation QA comes out negative (we tried to cross the left edge of the lattice in Fig.4.3). In this case it is again necessary to put $x_{0.3} = 3\Delta$ (return to the right edge of the lattice, and to change $x_{0.2}$ by Δ. These two steps are accomplished according to instructions RA and SA. Before starting to execute the instruction SA, however, it is tried (instruction RB) whether transgression of the lower edge of the lattice is likely to occur. Problem III is solved, and the result of the operation according to RB shows that further execution of instruction SA would be attended by transgression of the lower edge of the lattice.

4.20. Problem IV

Given is the entire optical system with its constants as before. To be solved is problem III for all of the directions a_0 of the entering ray selected in the

manner described in paragraph 4.5. Before exit of the result of every problem III, the value q_2 must exit, which is characteristic of the considered direction.

Problem IV is solved with the entire described instructional network (at which instructions VA, VB are disregarded).

All of the given values are stored in the same memories as in problem III, but the constants $a_{0.1}$, $a_{0.2}$, giving the directions of the entering ray, are not yet set. They must first be calculated from the values p_1 and q_2. The constant p_1 is stored in memory g0. The value q_2 is in memory k0, and is at the beginning equal to five.

The instructions of group A effect, on the one hand, the entrance of the value q_2, and on the other hand the calculation and correct storage of the values $a_{0.1}$, $a_{0.2}$.

This is the beginning for the solution of problem III for the first direction. After this problem has been solved, instruction TA is valid. The purpose of this instruction is to reach the state where the values 3Δ, 3Δ are in memories m0, m1. This is necessary in order to be able again to begin the solution of problem III for the next direction. The change in the direction is tried with instruction UA by changing the value q_2 by unity. Then it is possible to return to instruction AA. Problem IV is completely solved when at execution of instruction UA 2 this latter comes out negative. Then the machine stops.

4.21. Remarks on Instructions VA and VB

These instructions effect that at the exit the resultant coordinate is perforated into the card from memory g11. This is such a large number that it cannot be regarded as the coordinate of the point in the focal plane. The presence of this number in the card indicates that the considered ray did not reach the focal plane.

---

*The content of memory g11 must be replaced in the instruction network for still other purposes.

STAT

The instructions of group V are valid whenever any of the four cases occur mentioned in paragraphs 4.6, 4.8, 4.10 and 4.11.

The case of paragraph 4.6 occurs when the result of operation NV comes out negative. In the case of paragraph 4.8, the result of operation FE comes out negative. The case of paragraph 4.10 occurs when the result of operation FZ or GA comes out negative. In the case of paragraph 4.11 the result of operation NL comes out negative.

This completes the description of the entire instructional network.

4.22. Summary

The preparation of such an instructional network requires considerable effort, but it must be remembered that the work is rather permanent and that the network can be used for investigating any centered optical system*. Before beginning the solution of the problem, the information has been stored in the memories of the machines, where not only the constants but also the entire instructional network is perforated into cards. The entire batch of these cards is kept after the solution of the problem. If the same problem comes up later for another centered optical system, all that is necessary is to exchange in the corresponding batch of cards those containing the constants of the old system with those containing the constants of the new system. The entire batch is placed into the machine**, and the machine can be started.

The cards with the results come out of the machine in the following order: The first card contains perforated the constant $q_2 = 5$, denoting the common entering

---

* Here we have only systems with spherical or plane boundaries. However, it should be noted that it is always possible to prepare the instructional network in such a way that it can also be used for systems containing, for example, also paraboloid boundaries.

** Even regardless of the sequence of the cards in the batch.

direction of the first 28 rays. The next 56 cards contain the numerical values of the coordinates x_2, x_3 of the point in which these rays intersect the focal plane. For each of these rays the corresponding value x_2 is perforated in one card and the value x_3 in the next card. The order of the individual rays of the considered direction is given by the order of the starting points in Fig.4.3. Perforated into 58 cards is the value $q_2 = 4$, indicating that the values of the coordinates in the next 56 cards belong to the rays of the second entering direction. After these 56 cards follow the cards with the value $q_2 = 3$, etc.

All cards containing the results coming from the machine are given a serial number, so that when it becomes necessary later to use some of the cards there is no danger of confusing the results. The cards with the results can be used as the basis for further mechanical processing.

Problem: Optical System

Model 1

Analysis	Vocabulary	Instruction: Index	Instruction: Operational Symbol	+	−	Remarks
	$0 = \langle 0\rangle$	!Q0	$0 \to \langle 0\rangle$			
	$1 = \langle 1\rangle$	!Q1	$1 \to \langle 1\rangle$			
	$R_1 = \langle p0\rangle$	!Q2	$R_1 \to \langle p0\rangle$			
	$s_{1,1} = \langle p1\rangle$	!Q3	$s_{1,1} \to \langle p1\rangle$			
	$N_1 = \langle p2\rangle$	!Q4	$N_1 \to \langle p2\rangle$			
	⋮		⋮			
	$p_1 = \langle g0\rangle$	!Q5	$p_1 \to \langle g0\rangle$			As the initial value q2, the number 5 is set.
	$q_2 = \langle k0\rangle$	!Q6	$5 \to \langle k0\rangle$			
	$A = \langle g1\rangle$	!Q7	$A \to \langle g1\rangle$			
		AA	$H2\langle k0\rangle \cdot 0 \cdot h0$	AB	AB	
		AB	$\langle g0\rangle \cdot \langle g0\rangle \cdot \langle h0\rangle$	AC	AC	
$a_{0,1} = \dfrac{p_1}{\sqrt{p_1^2+q_2^2}}$		AC	$\langle k0\rangle \cdot \langle k0\rangle \cdot \langle h1\rangle$	AD	AD	
$a_{0,2} = \dfrac{-q_2}{\sqrt{p_1^2+q_2^2}}$	$p_1^2+q_2^2 = \langle h0\rangle$	AD	$\langle h0\rangle \cdot \langle h1\rangle \cdot h0$	AE	AE	The first approximation is set as $u_1 = 11$.
	$u_n = \langle h13\rangle$	!Q8	$11 \to \langle h13\rangle$			
		!Q9	$\varepsilon_1 \to \langle g4\rangle$			
$a_{0,3} = 0$	$0.5 = \langle g14\rangle$	!Q10	$0.5 \to \langle g14\rangle$			
		AE	$\langle h0\rangle \cdot \langle h13\rangle \cdot h1$	AF	AF	
$\left\lvert \dfrac{p_1^2+q_2^2}{u_n} - u_n \right\rvert < \varepsilon_1$?		AF	$\langle h13\rangle \cdot \langle h1\rangle \cdot h2$	AG	AG	
		AG	$\langle g4\rangle \cdot \langle h2\rangle \cdot h2$	AH	AL	
$u_{n+1} = 0.5\left(\dfrac{p_1^2+q_2^2}{u_n} + u_n\right)$		AH	$\langle h1\rangle \cdot \langle h13\rangle \cdot h1$	AK	AK	

Fig.4.7

Analysis	Vocabulary	Instruction: Index	Instruction: Operational Symbol	+	−	Remarks
$u_n \to \sqrt{p_1^2 + q_2^2}$	$u_{n+1} \to \langle h13 \rangle$	AK	$\langle g14 \rangle \cdot \langle h1 \rangle \to \langle h13 \rangle$	AE	AE	
	$\sqrt{p_1^2 + q_2^2} \to \langle h13 \rangle$					
	$a_{0,1} \to \langle k1 \rangle$	AL	$\langle g0 \rangle : \langle h13 \rangle \to \langle k1 \rangle$	AM	AM	
	$a_{0,2} \to \langle k2 \rangle$	AM	$-\langle k0 \rangle : \langle h13 \rangle \to \langle k2 \rangle$	BA	BA	Set $x_{0.2}$ "
	$x_{0,2} \to \langle m0 \rangle$	!Q11	$3.1 \to \langle m0 \rangle$			
	$x_{0,3} = \langle m1 \rangle$	!Q12	$3.4 \to \langle m1 \rangle$			" $= x_{0.3}$ "
		BA	$\langle k1 \rangle + \langle 0 \rangle \to \langle n3 \rangle$	BB	BB	3 Δ
		BB	$\langle k2 \rangle + \langle 0 \rangle \to \langle n4 \rangle$	BC	BC	
		BC	$\langle 0 \rangle + \langle 0 \rangle \to \langle n5 \rangle$	BD	BD	
		BD	$\langle 0 \rangle + \langle 0 \rangle \to \langle n6 \rangle$	BE	BE	
		BE	$\langle m0 \rangle + \langle 0 \rangle \to \langle n7 \rangle$	BF	BF	
		BF	$\langle m1 \rangle + \langle 0 \rangle \to \langle n10 \rangle$	CA	CA	
	$p0 \to \langle g11 \rangle$	!Q13	$p0 \to \langle g11 \rangle$			Correct Addresses "i" in Instructions DA, DB, JA
	$p1 \to \langle g12 \rangle$	!Q14	$p1 \to \langle g12 \rangle$			
	$p2 \to \langle g13 \rangle$	!Q15	$p2 \to \langle g13 \rangle$			
		CA	$\langle g11 \rangle = C!\langle DA + 1 \rangle \to !\langle DA + 1 \rangle$	CB	CB	
		CB	$\langle g12 \rangle = C!\langle DB + 1 \rangle \to !\langle DB + 1 \rangle$	CC	CC	
		CC	$\langle g13 \rangle = C!\langle JA + 1 \rangle \to !\langle JA + 1 \rangle$	DA	DA	Addresses p0, p1 were reached with instructions DA, DB at execution of instructions CA, CB
		DA	$\langle p0 \rangle + \langle 0 \rangle \to \langle n0 \rangle$	DB	DB	
		DB	$\langle p1 \rangle + \langle 0 \rangle \to \langle n1 \rangle$	EA	EA	
		EA	$-\langle g4 \rangle + \lvert\langle n0 \rangle\rvert \to \langle h0 \rangle$	FA	GA	
$\lvert R_j \rvert - \varepsilon < 0$? [Expression (4,15)]	$R_j \to \langle n0 \rangle$					
	$s_{j,1} \to \langle n1 \rangle$					
	$a_{j-1,1} \to \langle n3 \rangle$					
	$a_{j-1,2} \to \langle n4 \rangle$					

Fig. 4.8

STAT

Model 1

Analysis	Vocabulary	Instruction: Index	Instruction: Operational Symbol	+	−	Remarks
Procedure for Sphere: $A_j = a_{j-1,3}(x_{j-1,1} - s_{j,1}) +$ $+ a_{j-1,2}x_{j-1,2} + a_{j-1,3}x_{j-1,3}$ $B_j = x^2_{j-1,1} + x^2_{j-1,2} + x^2_{j-1,3} -$ $- 2x_{j-1,1}s_{j,1} + s^2_{j,1} - R^2_j =$ $= [(x_{j-1,1} - s_{j,1}) + R_j] \cdot$ $\cdot [(x_{j-1,1} - s_{j,1}) - R_j] +$ $+ x^2_{j-1,2} + x^2_{j-1,3}$	$a_{j-1,3} = \langle n5\rangle$ $x_{j-1,1} = \langle n6\rangle$ $x_{j-1,2} = \langle n7\rangle$ $x_{j-1,3} = \langle n10\rangle$ $x_{j-1,1} - s_{j,1} = \langle h0\rangle$	FA	$-\langle n1\rangle + \langle n6\rangle \to \langle h0\rangle$	FB	FB	
		FB	$\langle n3\rangle \cdot \langle h0\rangle \to \langle h1\rangle$	FC	FC	
		FC	$\langle n4\rangle \cdot \langle n7\rangle \to \langle h2\rangle$	FD	FD	
		FD	$\langle h1\rangle + \langle h2\rangle \to \langle h1\rangle$	FE	FE	
		FE	$\langle n5\rangle \cdot \langle n10\rangle \to \langle h2\rangle$	FF	FF	
	$A_j = \langle h7\rangle$	FF	$\langle h1\rangle + \langle h2\rangle \to \langle h7\rangle$	FG	FG	
		FG	$\langle n0\rangle + \langle h0\rangle \to \langle h1\rangle$	FH	FH	
		FH	$-\langle n0\rangle + \langle h0\rangle \to \langle h2\rangle$	FJ	FJ	
		FJ	$\langle h1\rangle \cdot \langle h2\rangle \to \langle h0\rangle$	FK	FK	
		FK	$\langle n7\rangle \cdot \langle n7\rangle \to \langle h1\rangle$	FL	FL	
		FL	$\langle h0\rangle + \langle h1\rangle \to \langle h0\rangle$	FM	FM	
		FM	$\langle n10\rangle \cdot \langle n10\rangle \to \langle h1\rangle$	FN	FN	
	$B_j = \langle h6\rangle$	FN	$\langle h0\rangle + \langle h1\rangle \to \langle h6\rangle$	FO	FO	
		FO	$\langle h7\rangle \cdot \langle h7\rangle \to \langle h0\rangle$	FP	FP	
	$A^2_j - B_j = \langle h0\rangle$	FP	$\langle h6\rangle + \langle h0\rangle \to \langle h0\rangle$	FQ	FA	Set as first approximation is $u_1 = 50$
	$u_n = \langle h14\rangle$	!Q16	$50 \to \langle h14\rangle$			
		!Q17	$\varepsilon_2 \to \langle g5\rangle$			
	$0{,}5 = \langle g14\rangle$					
		FQ	$\langle h0\rangle : \langle h14\rangle \to \langle h1\rangle$	FR	FR	
$\left\lvert \frac{A^2_j - B_j}{u_n} - u_n \right\rvert < \varepsilon_2\,?$		FR	$-\langle h14\rangle + \langle h1\rangle \to \langle h2\rangle$	FS	FS	
		FS	$\langle g5\rangle \cdot \langle h2\rangle \to \langle h2\rangle$	FT	FX	
$u_{n+1} = 0{,}5\left(\frac{A^2_j - B_j}{u_n} + u_n\right)$		FT	$\langle h1\rangle + \langle h14\rangle \to \langle h1\rangle$	FU	FU	
$u_n \to \sqrt{A^2_j - B_j}$	$u_{n+1} = \langle h14\rangle$	FU	$\langle g14\rangle \cdot \langle h1\rangle \to \langle h14\rangle$	FQ	FQ	

Fig.4.9

STAT

Model 1

Analysis	Vocabulary	Instruction: Index	Instruction: Operational Symbol	Instruction: +	Instruction: −	Remarks
	$\sqrt{A_j^2 - B_j} \cdot \langle h14\rangle$	FX	$\mathrm{sgn}\,\langle n0\rangle \to \langle h0\rangle$	FY	FY	
		FY	$\langle h0\rangle\ \mathrm{sgn}\,\langle h14\rangle \to \langle h0\rangle$	FZ	FZ	
		FZ	$\langle h7\rangle + \langle h0\rangle \to \langle h7\rangle$	HA	VA	
		GA	$-\langle n0\rangle \div \langle n1\rangle \to \langle h0\rangle$	GB	VA	
		GB	$\langle h0\rangle : \langle n3\rangle \to \langle h7\rangle$	HA	HA	
Procedure for Plane:	$t_j \to \langle h7\rangle$					
$t_j = \dfrac{s_{j,1} - x_{j-1,1}}{a_{j-1,1}}$ [Expression (4.12)]	$a_{j-1,1} \to \langle n3\rangle$					
	$a_{j-1,2} \to \langle n4\rangle$					
	$a_{j-1,3} \to \langle n5\rangle$					
	$x_{j-1,1} \to \langle n6\rangle$					
	$x_{j-1,2} \to \langle n7\rangle$					
	$x_{j-1,3} \to \langle n10\rangle$	HA	$\langle n3\rangle \cdot \langle h7\rangle \to \langle h0\rangle$	HB	HB	
$x_{j,1} = x_{j-1,1} + a_{j-1,1}t_j$	$x_{j,1}\ \langle n6\rangle$	HB	$\langle n6\rangle + \langle h0\rangle \to \langle n6\rangle$	HC	HC	
		HC	$\langle n4\rangle \cdot \langle h7\rangle \to \langle h0\rangle$	HD	HD	
$x_{j,2} = x_{j-1,2} + a_{j-1,2}t_j$	$x_{j,2}\ \langle n7\rangle$	HD	$\langle n7\rangle + \langle h0\rangle \to \langle n7\rangle$	HE	HE	
		HE	$\langle n5\rangle \cdot \langle h7\rangle \to \langle h0\rangle$	HF	HF	
$x_{j,3} = x_{j-1,3} + a_{j-1,3}t_j$	$x_{j,3}\ \langle n10\rangle$	HF	$\langle n10\rangle + \langle h0\rangle \to \langle n10\rangle$	JA	JA	
	$N_j\ \langle n2\rangle$	JA	$\langle p2\rangle : \langle 0\rangle \to \langle n2\rangle$	KA	PA	
$\lvert R_j\rvert - \varepsilon < 0$?		KA	$\langle g4\rangle + \lvert\langle n0\rangle\rvert \to \langle h0\rangle$	LA	MA	Address p2 was reached with instruction JA in execution of instruction CC
[Expression (4.15)]	$x_{j,1} \to \langle n6\rangle$					
	$x_{j,2}\ \langle n7\rangle$					
	$x_{j,3} \to \langle n10\rangle$					
	$R_j \to \langle n0\rangle$					
	$s_{j,1} \to \langle n1\rangle$	LA	$\langle n1\rangle - \langle n6\rangle \to \langle h0\rangle$	LB	LB	
Procedure for Sphere:	$n_{j,1}\ \langle h10\rangle$	LB	$\langle h0\rangle : \langle n0\rangle \to \langle h10\rangle$	LC	LC	
$n_{j,1} = (x_{j,1} - s_{j,1}) : R_j$	$n_{j,2}\ \langle h11\rangle$	LC	$\langle n7\rangle : \langle n0\rangle \to \langle h11\rangle$	LD	LD	
$n_{j,2} = x_{j,2} : R_j$	$n_{j,3} \to \langle h12\rangle$	LD	$\langle n10\rangle : \langle n0\rangle \to \langle h12\rangle$	NA	NA	
$n_{j,3} = x_{j,3} : R_j$ [Expression (4.3)]						

Fig.4.10

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

Analysis	Vocabulary	Instruction: Index	Instruction: Operational Symbol	+	−	Remarks
Procedure for Plane:		MA	$\langle 1\rangle + \langle 0\rangle \rightarrow \langle h10\rangle$	MB	MB	
$n_{j,1} = 1$		MB	$\langle 0\rangle + \langle 0\rangle \rightarrow \langle h11\rangle$	MC	MC	
$n_{j,2} = 0$		MC	$\langle 0\rangle + \langle 0\rangle \rightarrow \langle h12\rangle$	NA	NA	
$n_{j,3} = 0$	$N_j \quad \langle n2\rangle$					
	$a_{j-1,1} = \langle n3\rangle$					
	$a_{j-1,2} = \langle n4\rangle$					
	$a_{j-1,3} = \langle n5\rangle$					
	$n_{j,1} = \langle h10\rangle$					
	$n_{j,2} = \langle h11\rangle$					
	$n_{j,3} = \langle h12\rangle$					
$O_j = N_j$ [eq.(4.9)]	$N_j \cdot a_{j-1,1} = \langle n3\rangle$	NA	$\langle n2\rangle \cdot \langle n3\rangle \rightarrow \langle n3\rangle$	NB	NB	Since vector a_{j-1} is no longer needed, it must be replaced by vector $N_j a_{j-1}$
$N_j \cos\alpha_j = (N_j a_{j-1}) \cdot n_j$	$N_j \cdot a_{j-1,2} = \langle n4\rangle$	NB	$\langle n2\rangle \cdot \langle n4\rangle \rightarrow \langle n4\rangle$	NC	NC	
	$N_j \cdot a_{j-1,3} = \langle n5\rangle$	NC	$\langle n2\rangle \cdot \langle n5\rangle \rightarrow \langle n5\rangle$	ND	ND	
		ND	$\langle n3\rangle \cdot \langle h10\rangle \rightarrow \langle h0\rangle$	NE	NE	
		NE	$\langle n4\rangle \cdot \langle h11\rangle \rightarrow \langle h1\rangle$	NF	NF	
		NF	$\langle h0\rangle + \langle h1\rangle \rightarrow \langle h0\rangle$	NG	NG	
		NG	$\langle n5\rangle \cdot \langle h12\rangle \rightarrow \langle h1\rangle$	NH	NH	
	$N_j \cdot \cos\alpha_j \rightarrow \langle h7\rangle$	NH	$\langle h0\rangle + \langle h1\rangle \rightarrow \langle h7\rangle$	NI	NI	
$K_j = +\sqrt{1 - N_j^2 + (N_j \cos\alpha_j)^2} -$		NI	$\langle n2\rangle \cdot \langle n2\rangle \rightarrow \langle h0\rangle$	NJ	NJ	
$- N_j \cos\alpha_j$ [Corrected eq. (4.11)]		NJ	$-\langle 1\rangle + \langle h0\rangle \rightarrow \langle h0\rangle$	NK	NK	
		NK	$\langle h7\rangle \cdot \langle h7\rangle \rightarrow \langle h1\rangle$	NL	NL	
$-1 + N_j^2 - (N_j \cos\alpha_j)^2 = -D_j$	$-D_j \rightarrow \langle h0\rangle$	NL	$-\langle h1\rangle + \langle h0\rangle \rightarrow \langle h0\rangle$	VA	NM	
	$u_n = \langle h15\rangle$	!Q18	$1 \rightarrow \langle h15\rangle$			Set as a first approximation is $u_1 = 1$
		!Q19	$\varepsilon_3 \rightarrow \langle g0\rangle$			
	$0{,}5 \rightarrow \langle g14\rangle$					
		NM	$-\langle h0\rangle : \langle h15\rangle \rightarrow \langle h1\rangle$	NN	NN	
$\left\| -\dfrac{-D_j}{u_n} - u_n \right\| < \varepsilon_3\,?$		NN	$\|-\langle h15\rangle : \langle h1\rangle\| \rightarrow \langle h2\rangle$	NO	NO	

Fig.4.11

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

POOR ORIGINAL

Model 1

Analysis	Vocabulary	Instruction: Index	Instruction: Operational Symbol	+	−	Remarks
$u_{n+1} = 0,5\left(-\frac{-D_j}{u_n} + u_n\right)$		NO	$-\langle g6\rangle + \langle h2\rangle \rightarrow \langle h2\rangle$	NP	NS	
$u_n \rightarrow \sqrt{D_j}$	$u_{n+1} = \langle h15\rangle$	NP	$\langle h1\rangle + \langle h15\rangle \rightarrow \langle h1\rangle$	NQ	NQ	
	$\sqrt{D_j} = \langle h15\rangle$	NQ	$\langle g14\rangle \cdot \langle h1\rangle \rightarrow \langle h15\rangle$	NM	NM	
	$K_j = \langle h5\rangle$	NS	$-\langle h7\rangle + \langle h15\rangle \rightarrow \langle h5\rangle$	NT	NT	
$e_j = N_j a_{j-1} + K_j n_j$		NT	$\langle h5\rangle \cdot \langle h10\rangle \rightarrow \langle h0\rangle$	NU	NU	
[satisfied by eq.(4.4)]	$a_{j,1} = \langle n3\rangle$	NU	$\langle n3\rangle + \langle h0\rangle \rightarrow \langle n3\rangle$	NV	NV	
$a_{j,1} < r_4$?		1Q20	$r_4 \rightarrow \langle g7\rangle$			
[inequality 4.14]		NV	$\langle g7\rangle + \langle n3\rangle \rightarrow \langle h0\rangle$	NW	VA	
		NW	$\langle h5\rangle \cdot \langle h11\rangle \rightarrow \langle h0\rangle$	NX	NX	
	$a_{j,2} = \langle n4\rangle$	NX	$\langle n4\rangle + \langle h0\rangle \rightarrow \langle n4\rangle$	NY	NY	
		NY	$\langle h5\rangle \cdot \langle h12\rangle \rightarrow \langle h0\rangle$	NZ	NZ	
	$a_{j,3} = \langle n5\rangle$	NZ	$\langle n5\rangle + \langle h0\rangle \rightarrow \langle n5\rangle$	OA	OA	
	$3 = \langle g10\rangle$	1Q21	$3 \rightarrow \langle g10\rangle$			Increase addresses "i" in instructions DA, BD, JA by 3
		OA	$C\langle DA+1\rangle + \langle g10\rangle \rightarrow \langle DA+1\rangle$	OB	OB	
		OB	$C\langle DB+1\rangle + \langle g10\rangle \rightarrow \langle DB+1\rangle$	OC	OC	
		OC	$C\langle JA+1\rangle + \langle g10\rangle \rightarrow \langle JA+1\rangle$	DA	DA	
	$x_{m,2} = \langle n7\rangle$					
	$x_{m,3} = \langle n10\rangle$	PA	$H2\langle n7\rangle + \langle 0\rangle \rightarrow \langle h0\rangle$	PB	PB	
		PB	$H2\langle 10\rangle + \langle 0\rangle \rightarrow \langle h0\rangle$	QA	QA	
	$x_{0,2} = \langle m0\rangle$					
	$x_{0,3} = \langle m1\rangle$					
	$A = \langle g1\rangle$					
	$2A = \langle g2\rangle$	1Q22	$2A \rightarrow \langle g2\rangle$			
	$3.1 = \langle g3\rangle$	1Q23	$3.1 \rightarrow \langle g3\rangle$			
		QA	$-\langle g1\rangle + \langle m1\rangle \rightarrow \langle m1\rangle$	BA	BA	

Fig. 4.12

STAT

Model 1

Analysis	Vocabulary	Instruction: Index	Instruction: Operational Symbol	+	−	Remarks
		RA	$\langle g3\rangle + \langle 0\rangle \rightarrow \langle m1\rangle$	RB	RB	
		RB	$\langle m0\rangle + \langle g2\rangle \rightarrow \langle h0\rangle$	SA	TA	
		SA	$-\langle g1\rangle + \langle m0\rangle \rightarrow \langle m0\rangle$	BA	BA	
		TA	$\langle g3\rangle + \langle 0\rangle \rightarrow \langle m0\rangle$	UA	UA	
	$q_2 = \langle h0\rangle$	UA	$-\langle 1\rangle + \langle h0\rangle \rightarrow \langle h0\rangle$	AA	ZA	
		VA	$H2\langle g11\rangle + \langle 0\rangle \rightarrow \langle h0\rangle$	VB	VB	
		VB	$H2\langle g11\rangle + \langle 0\rangle \rightarrow \langle h0\rangle$	QA	QA	
		ZA	STOP			

Fig. 4.13

Problem: Optical System

Model 2

Index	Address		Entering Information		Altered Information	Remarks
!Q0	0000	P	.00000000 0 00 0	B	0	
!Q1	0001		.40000000 0 01 0	B	1	
AA	0002		0420 0004 S			
	0003	P	0000 0004 0440 H2			
AB	0004		0420 0006 N			
	0005	P	0400 0006 0400			
AC	0006	P	0421 0010 N			
	0007		0440 0010 0440			
AD	0010		0420 0012 S			
	0011	P	0421 0012 0420			
AE	0012	P	0121 0014 D			
	0013		0433 0014 0420			
AF	0014	P	0422 0016 KSM			
	0015		0421 0016 0433			
AG	0016		0422 0020 SM			
	0017	P	0422 0024 0404			
AH	0020		0421 0022 S			
	0021	P	0433 0022 0121			
AK	0022	P	0433 0012 N			
	0023		0121 0012 0414			
AL	0024	P	0441 0026 D			
	0025		0433 0026 0400			
AM	0026		0442 0030 DM			
	0027	P	0433 0030 0440			
BA	0030	P	0463 0032 S			
	0031		0000 0032 0441			
BB	0032		0464 0034 S			
	0033	P	0000 0034 0442			
BC	0034		0465 0036 S			
	0035	P	0000 0036 0000			
BD	0036	P	0466 0040 S			
	0037		0000 0040 0000			

Fig. 4.14

Index	Address		Entering Information	Altered Information	Remarks
BE	0040		0467 0042 S		
	0041	P	0000 0042 0450		
BF	0042	P	0470 0044 S		
	0043		0600 0044 0451		
CA	0044	P	0053 0046 WY		
	0045		0053 0046 0411		
CB	0046		0055 0050 WY		
	0047	P	0055 0050 0412		
CC	0050	P	0157 0052 WY		
	0051		0157 0052 0413		
DA	0052		0460 0054 S		
	0053	P	0000 0054 --	0000 0054 0500 etc	
DB	0054		0461 0056 S		
	0055	P	0000 0056 -	0000 0056 0501 etc	
EA	0056	P	0420 0060 JSM		
	0057		0460 0136 0404		
FA	0060	P	0420 0062 SM		
	0061		0466 0062 0461		
FB	0062		0421 0064 N		
	0063	P	0420 0064 0403		
FC	0064		0422 0066 N		
	0065	P	0467 0066 0464		
FD	0066	P	0421 0070 S		
	0067		0422 0070 0421		
FE	0070		0422 0072 N		
	0071	P	0470 0072 0465		
FF	0072	P	0427 0074 S		
	0073		0422 0074 0421		
FG	0074	P	0421 0076 S		
	0075		0420 0076 0460		
FH	0076		0122 0100 SM		
	0077	P	0420 0100 0460		

Fig. 4.15

STAT

Problem: Optical System

Model 2

Index	Address		Entering Information	Altering Information	Remarks
FJ	0100		0420 0102 N		
	0101	P	0422 0102 0421		
FK	0102	P	0421 0104 N		
	0103		0467 0104 0467		
FL	0104	P	0420 0106 S		
	0105		0421 0106 0420		
FM	0106		0421 0110 N		
	0107	P	0470 0110 0470		
FN	0110	P	0426 0112 S		
	0111		0421 0112 0420		
FO	0112		0420 0114 N		
	0113	P	0427 0114 0427		
FP	0114		0420 0116 SM		
	0115	P	0420 0310 0426		
FQ	0116	P	0421 0120 D		
	0117		0434 0120 0420		
FR	0120	P	0422 0122 KSM		
	0121		0421 0122 0434		
FS	0122		0422 0124 SM		
	0123	P	0422 0130 0405		
FT	0124		0421 0126 S		
	0125	P	0434 0126 0421		
FU	0126	P	0434 0116 N		
	0127		0421 0116 0414		
FX	0130		0420 0132 SYZ		
	0131	P	0460 0132 0000		
FY	0132	P	0420 0134 WYZ		
	0133		0434 0134 0420		
FZ	0134	P	0427 0142 SM		
	0135		0420 0310 0427		
GA	0136		0420 0140 SM		
	0137	P	0401 0310 0466		

Fig. 4.16

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

Model 2

Index	Address		Entering Information	Altered Information	Remarks
HH	0140	P	0427 0142 D		
	0141		0463 0142 0420		
HA	0142		0420 0144 N		
	0143	P	0427 0144 0463		
HB	0144		0466 0146 S		
	0145	P	0420 0146 0466		
HC	0146	P	0420 0150 N		
	0147		0427 0150 0464		
HD	0150		0467 0152 S		
	0151	P	0420 0152 0467		
HE	0152	P	0420 0154 N		
	0153		0427 0154 0465		
HF	0154	P	0470 0156 S		
	0155		0420 0156 0470		
JA	0156		0462 0160 S		
	0157	P	0000 0270 —	0000 0270 0502 etc	
KA	0160		0420 0162 JSM		
	0161	P	0460 0172 0404		
LA	0162	P	0420 0164 SM		
	0163		0466 0164 0461		
LB	0164	P	0430 0166 D		
	0165		0460 0166 0420		
LC	0166		0431 0170 D		
	0167	P	0460 0170 0467		
LD	0170	P	0432 0200 D		
	0171		0460 0200 0470		
MA	0172		0430 0174 S		
	0173	P	0000 0174 0001		
MB	0174		0431 0176 S		
	0175	P	0000 0176 0000		
MC	0176	P	0432 0200 S		
	0177		0000 0200 0000		

Fig. 4.17

[redacted]

STAT

Model 2

Index	Address		Entering Information	Altered Information	Remarks
NA	0200		0463 0202 N		
	0201	P	0463 0202 0462		
NB	0202	P	0464 0204 N		
	0203		0464 0204 0462		
NC	0204	P	0465 0206 N		
	0205		0465 0206 0462		
ND	0206		0420 0210 N		
	0207	P	0430 0210 0463		
NE	0210	P	0421 0212 N		
	0211		0431 0212 0464		
NF	0212		0420 0214 S		
	0213	P	0421 0214 0420		
NG	0214		0421 0216 N		
	0215	P	0432 0216 0465		
NH	0216	P	0427 0220 S		
	0217		0421 0220 0420		
NI	0220	P	0420 0222 N		
	0221		0462 0222 0462		
NJ	0222		0420 0224 SM		
	0223	P	0420 0224 0001		
NK	0224		0421 0226 N		
	0225	P	0427 0226 0427		
NL	0226	P	0420 0310 SM		
	0227		0420 0230 0421		
NM	0230		0421 0232 DM		
	0231	P	0435 0232 0422		
NN	0232	P	0422 0234 KSM		
	0233		0421 0234 0435		
NO	0234	P	0422 0236 SM		
	0235		0422 0242 0400		
NP	0236		0421 0240 S		
	0237	P	0435 0240 0421		

Fig. 4.18

[redacted]

STAT

Model 2

Index	Address		Entering Information	Altered Information	Remarks
NQ	0240	P	0435 0230 N		
	0241		0421 0230 0414		
NS	0242		0425 0244 SM		
	0243	P	0435 0244 0427		
NT	0244		0420 0246 N		
	0245	P	0430 0246 0425		
NU	0246	P	0463 0250 S		
	0247		0420 0250 0463		
NV	0250		0420 0252 SM		
	0251	P	0463 0310 0407		
NW	0252	P	0420 0254 N		
	0253		0431 0254 0425		
NX	0254	P	0464 0256 S		
	0255		0420 0256 0404		
NY	0256		0420 0260 N		
	0257	P	0432 0260 0425		
NZ	0260		0463 0262 S		
	0261	P	0420 0262 0465		
OA	0262	P	0053 0264 SWY		
	0263		0053 0264 0410		
OB	0264	P	0055 0266 SWY		
	0265		0055 0266 0410		
OC	0266		0157 0052 SWY		
	0267	P	0157 0052 0410		
PA	0270	P	0420 0272 S		
	0271		0000 0272 0467 H2		
PB	0272		0420 0274 S		
	0273	P	0000 0274 0470 H2		
QA	0274		0451 0030 SM		
	0275	P	0451 0276 0401		
RA	0276	P	0451 0300 S		
	0277		0000 0300 0403		

Fig. 4.19

Model 2

Index	Address		Entering Information	Altered Information	Remarks
RB	0300	P	0420 0302 S		
	0301		0102 0304 0450		
SA	0302		0450 0030 SM		
	0303	P	0450 0030 0401		
TA	0304		0450 0306 S		
	0305	P	0000 0306 0403		
CA	0396	P	0440 0002 SM		
	0307		0440 0314 0001		
VA	0310		0420 0312 S		
	0311	P	0000 0312 0411 H2		
VB	0312	P	0420 0274 S		
	0313		0000 0274 0411 H2		
ZA	0314	P	0000 0000		
	0315		0000 0000 0000		
	0316				
	0317	P			
	0320				
	0321	P			
	0322	P			
	0323				
	0324	P			
	0325				
	0320				
	0327	P			
	0330	P			
	0331				
	0332				
	9333	P			
	0334				
	0335	P			
	0336	P			
	0337				

Fig. 4.20

[redacted]

STAT

Problem: Optical System

Model 2

Index	Address		Entering Information	Altered Information	Remarks
!Q5	0400		p_1		g0 400
!Q7	0401	P	1		g1 = 401
!Q22	0402	P	2,1		g2 = 402
!Q23	0403		3,4		g3 = 403
!Q9	0404	P	r_1		g4 404
!Q17	0405		r_2		g5 405
!Q19	0406		r_3		g6 406
!Q20	0407	P	r_4		g7 407
!Q21	0410	P	,60000000 0 02 0	= 3	g10 = 410
!Q13	0411		,50000000 0 11 0	= 500 p0	g11 = 411
!Q14	0412		,50100000 0 11 0	= 501 = p1	g12 = 412
!Q15	0413	P	,50200000 0 11 0	= 502 = p2	g13 = 413
!Q10	0414		,40000000 0 00 0	= 0.5	g14 = 414
	0415	P			
	0416	P			
	0417				
	0420	P			h0 = 420
	0421				h1 = 421
	0422			...	h2 = 422
	0423	P			
	0424				
	0425	P		K_j	h5 = 425
	0426	P		$B_j, -D_j$	h6 = 426
	0427			$A_j, t_j, N_j, \cos \alpha_j$	h7 = 427
	0430			$n_{j,1}$	h10 = 430
	0431	P		$n_{j,2}$	h11 = 431
	0432	P		$n_{j,3}$	h12 = 432
!Q8	0433		,54000000 0 04 0	= 11, $u_n \to \sqrt{p_1^2 + q_1^2}$	h13 = 433
!Q16	0434	P	,62000000 0 06 0	= 50, $u_n \to \sqrt{A_j^2 - B_j}$	h14 = 434
!Q18	0435		,40000000 0 01 0	= 1, $u_n \to \sqrt{D_j}$	h15 = 435
	0436				
	0437	P			

Fig. 4.21

Model 2 Problem: Optical System

Index	Address		Entering Information	Altered Information	Remarks	
Q8	0440	P	.50000000 0 03 0	$5. q_t$	l0	440
	0441			$a_{0,1}$	l1	441
	0442			$a_{0,2}$	l2	442
	0443	P				
	0444					
	0445	P				
	0446	P				
	0447					
Q11	0450		3 1	$x_{0,1}$	m0	450
Q12	0451	P	3 1	$x_{0,2}$	m1	451
	0452	P				
	0453					
	0454	P				
	0455					
	0456					
	0457	P				
	0460			R_j	n0	460
	0461	P		$s_{j,1}$	n1	461
	0462	P		N_j	n2	462
	0463			$a_{j-1,1}$	n3	463
	0464	P		$a_{j-1,2}$	n4	464
	0465			$a_{j-1,3}$	n5	465
	0466			$x_{j-1,1}$	n6	466
	0467	P		$x_{j-1,2}$	n7	467
	0470	P		$x_{j-1,3}$	n10	470
	0471					
	0472					
	0473	P				
	0474					
	0475	P				
	0476	P				
	0477					

Fig. 4.22

[redacted]

STAT

Model 2

Problem: Optical System

Index	Address		Entering Information	Altered Information	Remarks
!Q2	0300	P	R_1		$p0$ - 500
!Q3	0301		$O_{s,1}$		$p1$ - 501
!Q4	0302		N_1		$p2$ - 502
	0303	P	etc.		etc.
	0304				
	0305	P			
	0306	P			
	0307				
	0310				
	0311	P			
	0312	P			
	0313				
	0314	P			
	0315				
	0316				
	0317	P			
	0320				
	0321	P			
	0322	P			
	0323				
	0324	P			
	0325				
	0326				
	0327	P			
	0330	P			
	0331				
	0332				
	0333	P			
	0334				
	0335	P			
	0336	P			
	0337				

Fig. 4.23

CHAPTER V

SOLUTION OF CONVENTIONAL DIFFERENTIAL EQUATIONS OF THE 2ND ORDER WITH THE AUTOMATIC CALCULATOR

The following solutions of concrete problems by means of the computer are selected from the field of conventional differential equations. There are two such problems:

1. Investigation on a very simple example of the formulation of the instructional network for the solution of a differential equation;

2. Ascertainment of modifications enabling employment of proposed instructional network for another automatic calculation.

Purposely selected was a very simple example in order to avoid difficulties in grasping the essential matter. The solution is obtained by a more complicated method which can be readily applied to very complicated systems of differential equations.

Let us examine the solution of the equation:

$$\frac{d^2x}{dt^2} = P_3(x) - f(t),$$

where

$$P_3(x) = a_1x^3 + a_2x^2 + a_3x + a_4, \qquad f(t) = \frac{b_1t + b_2}{c_1t^2 + c_2t + c_3}$$

satisfying the starting conditions

$$x = x_0, \ \frac{dx}{dt} = v_0 \quad \text{for } t = t_0$$

It is additionally assumed that the sought integral curve does not approach any of the singular points of this equation. This equation is easily transformed into the systems:

[redacted]

175

APPLICATION OF THE METHOD

5.1 Runge-Kutta's Method

For the solution, the Runge-Kutta method* is used because it requires a very small number of memories for the machine. In view of the simplicity of this case, this circumstance is rather unimportant since scarcely 10% of the memory capacity of the machine is utilized. A deficiency of memories manifests itself in the use of systems having a very large number of differential equations.

We select the corresponding increment h independent of the variation in t, and, beginning with the given initial conditions v_o, x_o, t_o (denoted v_{io}, x_{io}, t_{io}), we first calculate

$$\left.\begin{aligned} k_{i0} &= [P_3(x_{i0}) + f(t_{i0})] \cdot h\,, \\ l_{i0} &= v_{i0} \cdot h\,, \end{aligned}\right\} \qquad (5.1)$$

and from this we calculate

$$\left.\begin{aligned} k_{i1} &= [P_3(x_{i1}) + f(t_{i1})] \cdot h \\ l_{i1} &= v_{i1} \cdot h \end{aligned}\right\} \qquad (5.2)$$

where

$$v_{i1} = v_{i0} + \frac{k_{i0}}{2}, \qquad x_{i1} = x_{i0} + \frac{l_{i0}}{2}, \qquad t_{i1} = t_{i0} + \frac{h}{2}$$

With the help of these values we determine

$$\left.\begin{aligned} k_{i2} &= [P_3(x_{i2}) + f(t_{i2})] \cdot h \\ l_{i2} &= v_{i2} \cdot h \end{aligned}\right\}, \qquad (5.3)$$

where

$$v_{i2} = v_{i0} + \frac{k_{i1}}{2}, \qquad x_{i2} = x_{i0} + \frac{l_{i1}}{2}, \qquad t_{i2} = t_{i0} + \frac{h}{2}$$

and finally

$$\begin{aligned} k_{i3} &= [P_3(x_{i3}) + f(t_{i3})] \cdot h \\ l_{i3} &= v_{i3} \cdot h \end{aligned} \qquad (5.4)$$

* Laska-Hruska, Theory and Practice of Numerical Calculation

APPLICATION OF THE METHOD

5.1 Runge-Kutta's Method

For the solution, the Runge-Kutta method* is used because it requires a very small number of memories for the machine. In view of the simplicity of this case, this circumstance is rather unimportant since scarcely 10% of the memory capacity of the machine is utilized. A deficiency of memories manifests itself in the use of systems having a very large number of differential equations.

We select the corresponding increment h independent of the variation in t, and, beginning with the given initial conditions v_o, x_o, t_o (denoted v_{io}, x_{io}, t_{io}), we first calculate

$$\left.\begin{aligned} k_{i0} &= [P_3(x_{i0}) + f(t_{i0})] \,.\, h \,, \\ l_{i0} &= v_{i0} \,.\, h \,, \end{aligned}\right\} \qquad (5.1)$$

and fron this we calculate

$$\left.\begin{aligned} k_{i1} &= [P_3(x_{i1}) + f(t_{i1})] \,.\, h \\ l_{i1} &= v_{i1} \,.\, h \end{aligned}\right\} \qquad (5.2)$$

where

$$v_{i1} = v_{i0} + \frac{k_{i0}}{2}, \qquad x_{i1} = x_{i0} + \frac{l_{i0}}{2}, \qquad t_{i1} = t_{i0} + \frac{h}{2}$$

With the help of these values we determine

$$\left.\begin{aligned} k_{i2} &= [P_3(x_{i2}) + f(t_{i2})] \,.\, h \\ l_{i2} &= v_{i2} \,.\, h \end{aligned}\right\}, \qquad (5.3)$$

where

$$v_{i2} = v_{i0} + \frac{k_{i1}}{2}, \qquad x_{i2} = x_{i0} + \frac{l_{i1}}{2}, \qquad t_{i2} = t_{i0} + \frac{h}{2}$$

and finally

$$\begin{aligned} k_{i3} &= [P_3(x_{i3}) + f(t_{i3})] \,.\, h \\ l_{i3} &= v_{i3} \,.\, h \end{aligned} \qquad (5.4)$$

* Laska-Hruska, Theory and Practice of Numerical Calculation

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

where

$$v_{i3} = v_{i0} + k_{i2}, \quad x_{i3} = x_{i0} + l_{i2}, \quad t_{i3} = t_{i0} + h$$

The further integral points of the curve are then given by

$$\begin{aligned} v_{i+1,0} &= v_{i0} + \tfrac{1}{6}(k_{i0} + 2k_{i1} + 2k_{i2} + k_{i3}), \\ x_{i+1,0} &= x_{i0} + \tfrac{1}{6}(l_{i0} + 2l_{i1} + 2l_{i2} + l_{i3}), \\ t_{i+1,0} &= t_{i0} + h. \end{aligned} \qquad (5.5)$$

Using these values as the new starting conditions, we obtain by the further steps of the Runge-Kutta method the new integral points of the curve. This procedure is constantly repeated. Since the calculation of the expressions for

$$k_{ij}, \ l_{ij}$$

where

$$j = 0, 1, 2, 3$$

is the same, it can be carried out with the same part of the instructional network. It is sufficient if we are only concerned with the exchange of the numbers v_{ij}, x_{ij}, t_{ij} for $v_{i,j+1}$, $x_{i,j+1}$, $t_{i,j+1}$. On this part of the instructional network it will then be possible to establish the further operations, namely, the calculation of $v_{i+1,0}$, $x_{i+1,0}$, $t_{i+1,0}$ according to eq.(5.5). Such a network, however, would contain a relatively large number of instructions, and hence would require a large number of memories for storage of the intermediate results k_{ij}, l_{ij}. This increased demand on the memories would, of course, be a matter of indifference with this very simple system, but in the solution of systems with a large number of differential equations a deficiency of memories might occur.

5.2 Modification of the Runge-Kutta Method

Next let us try to reformulate the instructional network more economically. It is again required to calculate equations (5.1), (5.2), (5.3), (5.4) with the

STAT

same part of the instructional network. In the calculation, the numbers v_{ij}, x_{ij}, t_{ij} are exchanged. We assume that, for facilitating the calculation of the first two equations of (5.5), it is necessary to add some further instructions, i.e., apparently seven. We multiply in (5.1), (5.2), (5.3), (5.4) instead of the number h the numbers h/6, h/3, h/3, h/6, by which we get on the left side $k_{i0}/6$, $k_{i1}/3$, $k_{i2}/3$, $k_{i3}/6$. These need only be added instead of v_{i0} for getting the desired $v_{i+1,0}$. This addition is carried out in such a way that at the end of the instruction network, according to which $v_{i,j+1}$ is calculated, the instruction which retains this value is added. Between the individual calculations the starting values $v_{i,j+1}$ and $t_{i,j+1}$ must also be calculated with the help of k_{ij} for further calculation. Exactly the same considerations apply to l_{ij} and x_{ij}.

MECHANICAL SOLUTION OF PROBLEMS

5.3 Draft of the Instructional Network in General Form

Before beginning to solve the problem, the necessary constants and starting values v_{io}, x_{io}, t_{io} must be stored in the machine in the corresponding memories. These correspond to the tabulation for the calculation of the form. We carry out the calculation for j = 0, and in the groups of instructions A and B we calculate $f(t_{i,j})$ and $P_3(x_{i,j})$. The manner of constructing the instructional network for the calculation of these expressions need not be described here, since the mathematician will already be sufficiently familiar with this.

Instruction CA has the form of $P_3(x_{ij}) + f(t_{ij})$. With the groups of instructions D we calculate

$$\frac{k_{i0}}{6} = [P_3(x_{i0}) + f(t_{i0})]\, q_0\,, \qquad v_{i0} + \frac{k_{i0}}{6}\,. \qquad q_0 = \frac{h}{6}\,.$$

The group of instructions E gives

$$\frac{l_{i0}}{6} = v_{i0}\, q_0\,, \quad x_{i0} + \frac{l_{i0}}{6}$$

$$x_{i1} = x_{i0} + m_0 \frac{l_{i0}}{6}\,, \qquad v_{i1} = v_{i0} + m_0 \cdot \frac{k_{i0}}{6}\,, \qquad m_0 = 3\,.$$

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

Next we must carry out the calculation for $j = 1$. For the calculation of the form we must tabulate the new values. For the calculation by machine this is carried out with the instructions of group T. We come to instruction FA, which increases the address of the instruction by 2. This establishes the instruction by which the calculation is repeated. As soon as this instruction is reached for the second time, the machine is proceeding along the path $j = 2$. In the group of instructions beginning with TA $m_0 = 3$, this is exchanged for $m_1 = \frac{3}{2}$ and $q_0 = \frac{h}{6}$ for $q_1 = \frac{h}{3}$. In the group of instructions beginning with TC, the values change to $t_{i1} = t_{i0} + \frac{h}{2}$ regardless of the variables. It is therefore necessary to continue to calculate

$$\frac{k_{i1}}{3} = [P_3(x_{i1}) + f(t_{i1})]q_1 . \quad \left(v_{i0} + \frac{k_{i0}}{6}\right) + \frac{k_{i1}}{3} . \qquad q_1 = \frac{h}{3}$$

$$\frac{l_{i1}}{3} = v_{i1} \cdot q_1 . \quad \left(x_{i0} + \frac{l_{i0}}{6}\right) + \frac{l_{i1}}{3} .$$

$$x_{i2} = x_{i0} + m_1 \cdot \frac{l_{i1}}{3} . \qquad v_{i2} = v_{i0} + m_1 \cdot \frac{k_{i1}}{3} , \qquad m_1 = \frac{3}{2}$$

Next we carry out the tabulation with instruction TE along the path $j = 2$, in which $m_1 = \frac{3}{2}$ is exchanged for $m_1 = 3$. Regardless of the variable it remains unchanged, because $t_{i2} = t_{i0} + \frac{h}{2}$. We must therefore repeat the calculation of $f(t)$ and continue until in the group of instructions B

$$\frac{k_{i2}}{3} = [P_3(x_{i2}) + f(t_{i2})]\, q_2 , \quad \left(v_{i0} + \frac{k_{i0}}{6} + \frac{k_{i1}}{3}\right) + \frac{k_{i2}}{3} . \qquad q_2 = \frac{h}{3} .$$

$$\frac{l_{i2}}{3} = v_{i2} q_2 . \quad \left(x_{i0} + \frac{l_{i0}}{6} + \frac{l_{i1}}{3}\right) + \frac{l_{i2}}{3} .$$

$$x_{i3} = x_{i0} + m_2 \frac{l_{i2}}{3} . \qquad v_{i3} = v_{i0} + m_2 \cdot \frac{l_{i2}}{3} . \qquad m_2 = 3$$

In the further calculation we exchange in instruction TF $q_2 = \frac{h}{3}$ for $q_3 = \frac{h}{6}$ and change the values regardless of the variables in the group of instructions beginning with TC

STAT

$$k_{i3} = [P_3(x_{i3}) + f(t_{i3})]\, q_3\,. \qquad \left(v_{i0} + \frac{k_{i0}}{6} + \frac{k_{i1}}{3} + \frac{k_{i2}}{3}\right) + \frac{k_{i3}}{6} = v_{i+1,0}\,.$$

$$\frac{l_{i3}}{6} = v_{i3} q_3\,. \qquad \left(x_{i0} + \frac{l_{i0}}{6} + \frac{l_{i1}}{3} + \frac{l_{i2}}{3}\right) + \frac{l_{i3}}{6} = x_{i+1,0}\,.$$

The group of instructions beginning with TG furnishes the commands for perforating the numbers $v_{i+1,0}$, $x_{i+1,0}$, $t_{i+1,0}$, and at the same time tabulates them as the starting values for the next step of the Runge-Kutta method. Thus the actual induction from i to i + 1 is obtained. The group of instructions beginning with TG, however, must still correct the address of the instruction FA by which the instruction is set for the continuation in such a way that the machine continues along the path j = 1 when instruction FA comes up again. Since the solution of the equation is of interest to us in the particular range (t_o, t_{max}), included in this group is instruction TM, by which the sign of the difference $t_{io} - t_{max}$ is ascertained. As soon as it comes out positive, the machine has reached the limit of the range in which we are insterested, and it stops.

The whole calculation proceeding in the machine may be briefly stated in the form of the following equations:

$$\begin{aligned}
r_{ij} &= [P_3(x_{ij}) + f(t_{ij})] \cdot q_j \\
v_{i0} + \sum_{k=0}^{j} r_{ik} &= \left(v_{i0} + \sum_{k=0}^{j-1} r_{ik}\right) + r_{ij} \\
s_{ij} &= v_{ij} \cdot q_j \\
x_{i0} + \sum_{k=0}^{j} s_{ik} &= \left(x_{i0} + \sum_{k=0}^{j-1} s_{ik}\right) + s_{ij} \\
x_{i,j+1} &= x_{i0} + m_j s_{ij} \\
v_{i,j+1} &= v_{i0} + m_j r_{ij}
\end{aligned}$$

Before each calculation, it is necessary to replace in the corresponding memories the numbers q_{j-1}, m_{j-1} by the numbers q_j, m_j according to the tabulation in

STAT

Fig.5.1. This is repeated four times, i.e., for $j = 0, 1, 2, 3$. The developed diagram of the instructional network is represented in Fig.5.2. The proposed instructional network in a general form is presented in Figs.5.3, 5.4, and 5.5.

j	0	1	2	3
q_j	$\frac{h}{6}$	$\frac{h}{3}$	$\frac{h}{3}$	$\frac{h}{6}$
m_j	3	1	3	3

Fig.5.1

For better orientation it may be interesting to note that the calculation of this equation in 50 steps took about half an hour. As already mentioned in the beginning, we used the Runge-Kutta method in this case only for didactic reasons. If it had been the purpose to obtain the solution in the shortest possible time, a method of differentiation requiring a larger number of memories would have been selected. In this way the time required for calculating the same number of steps would have been reduced to about a third.

Finally, it may be interesting to note that recently a special method has been established for solving differential equations with the automatic computer*. It is also based on the Runge-Kutta method, and the calculation requires an even smaller number of memories than for the Runge-Kutta method as it has been employed here.

* Gill, A process for the step-by step integration of differential equations in an automatic digital computing machine

STAT

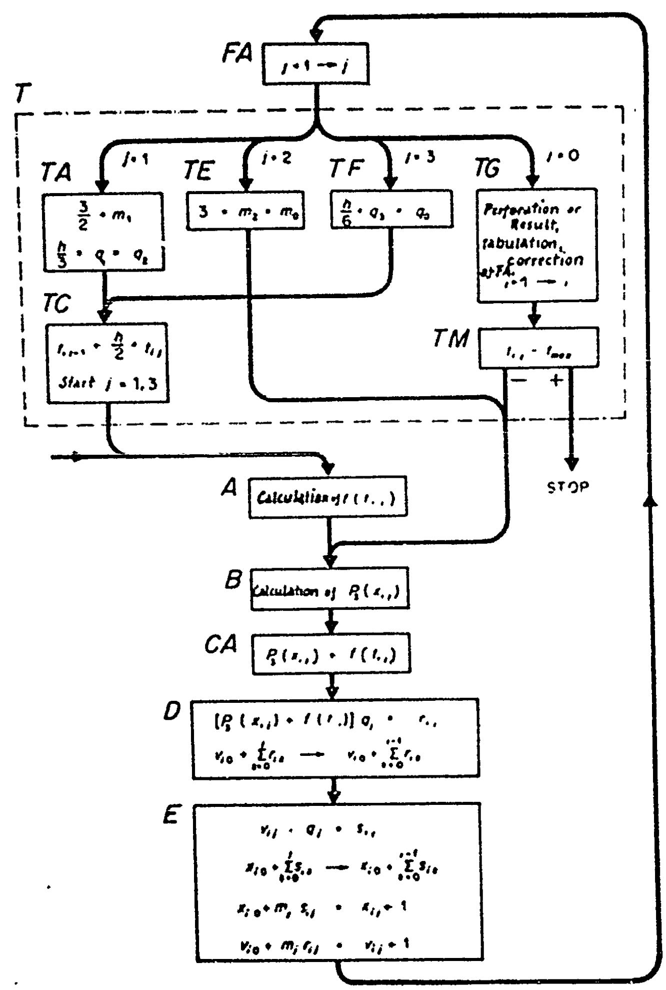


Fig.5.2 - Developed Diagram

Model 1 Problem: Differential Equation Page 1

Analysis	Instruction: Vocabulary	Index	Operational Symbol	+	−	Remarks
	$0 \to \langle 0 \rangle$	!Q0	$0 \to \langle 0 \rangle$			
	$1 \to \langle 1 \rangle$	!Q1	$1 \to \langle 1 \rangle$			
$t_0 = t_{00}$	$t_0 \to \langle p7 \rangle$	!Q2	$t_0 \to \langle p7 \rangle$			
	$r_0 \to \langle p4 \rangle$	!Q3	$r_0 \to \langle p4 \rangle$			
	$r_0 \to \langle p5 \rangle$	!Q4	$r_0 \to \langle p5 \rangle$			
	$r_0 \to \langle p6 \rangle$	!Q5	$r_0 \to \langle p6 \rangle$			
	$x_0 \to \langle p1 \rangle$	!Q6	$x_0 \to \langle p1 \rangle$			
	$x_0 \to \langle p2 \rangle$	!Q7	$x_0 \to \langle p2 \rangle$			
	$x_0 \to \langle p3 \rangle$	!Q8	$x_0 \to \langle p3 \rangle$			
	$b_1 \to \langle w11 \rangle$	!Q9	$b_1 \to \langle w11 \rangle$			
	$b_2 \to \langle w12 \rangle$	!Q10	$b_2 \to \langle w12 \rangle$			
	$c_1 \to \langle w13 \rangle$	!Q11	$c_1 \to \langle w13 \rangle$			
	$c_2 \to \langle w14 \rangle$	!Q12	$c_2 \to \langle w14 \rangle$			
	$c_3 \to \langle w15 \rangle$	!Q13	$c_3 \to \langle w15 \rangle$			
	$t_U \to \langle p7 \rangle$					
	$b_1 t_U \to \langle z1 \rangle$	AA	$\langle w11 \rangle \cdot \langle p7 \rangle \to \langle z1 \rangle$	AB	AB	
$b_1 t_U + b_2 = g_1$	$g_1 \to \langle z1 \rangle$	AB	$\langle z1 \rangle + \langle w12 \rangle \to \langle z1 \rangle$	AC	AC	
	$c_1 t_U \to \langle z2 \rangle$	AC	$\langle w13 \rangle \cdot \langle p7 \rangle \to \langle z2 \rangle$	AD	AD	
$c_1 t_U + c_2 = g_2$	$g_2 \to \langle z2 \rangle$	AD	$\langle z2 \rangle + \langle w14 \rangle \to \langle z2 \rangle$	AE	AE	
$(c_1 t_U + c_2)\, t_U = g_3$	$g_3 \to \langle z2 \rangle$	AE	$\langle z2 \rangle \cdot \langle p7 \rangle \to \langle z2 \rangle$	AF	AF	
$(c_1 t_U + c_2)\, t_U + c_3 = g_4$	$g_4 \to \langle z2 \rangle$	AF	$\langle z2 \rangle + \langle w15 \rangle \to \langle z2 \rangle$	AG	AG	
	$f(t) \to \langle z2 \rangle$	AG	$\langle z1 \rangle : \langle z2 \rangle \to \langle z2 \rangle$	BA	BA	
	$a_1 \to \langle w1 \rangle$	!Q14	$a_1 \to \langle w1 \rangle$			
	$a_2 \to \langle w2 \rangle$	!Q15	$a_2 \to \langle w2 \rangle$			
	$a_3 \to \langle w3 \rangle$	!Q16	$a_3 \to \langle w3 \rangle$			
	$a_4 \to \langle w4 \rangle$	!Q17	$a_4 \to \langle w4 \rangle$			
	$x_U \rho_1 \to \langle z1 \rangle$	BA	$\langle p2 \rangle \cdot \langle w1 \rangle \to \langle z1 \rangle$	BB	BB	
$x_U \rho_1 + a_2 = g_5$	$g_5 \to \langle z1 \rangle$	BB	$\langle z1 \rangle + \langle w2 \rangle \to \langle z1 \rangle$	BC	BC	
$g_5 x_U = g_6$	$g_6 \to \langle z1 \rangle$	BC	$\langle z1 \rangle \cdot \langle p2 \rangle \to \langle z1 \rangle$	BD	BD	

Fig.5.3.- Instructional Network for Solution of Differential Equation

Analysis	Vocabulary	Instruction: Index	Instruction: Operational Symbol			Remarks
$g_6 + a_3 = g_7$	$g_7 \to \langle z1 \rangle$	BD	$\langle z1 \rangle - \langle u3 \rangle \to \langle z1 \rangle$	BE	BE	
$g_7 \cdot x_{ij} = g_8$	$g_8 \to \langle z1 \rangle$	BE	$\langle z1 \rangle \cdot \langle p2 \rangle \to \langle z1 \rangle$	BF	BF	
$g_8 + a_4 = P_3(x_{ij})$	$P_3(x_{ij}) \to \langle z1 \rangle$	BF	$\langle z1 \rangle + \langle u4 \rangle \to \langle z1 \rangle$	CA	CA	
$P_3(x_{ij}) + f(t_{ij}) = g_9$	$g_9 \to \langle z1 \rangle$	CA	$\langle z1 \rangle + \langle z2 \rangle \to \langle z1 \rangle$	DA	DA	
$\begin{array}{c\|c\|c\|c\|c} j & 0 & 1 & 2 & 3 \\ \hline m_j & 3 & [illegible] & 3 & 3 \\ q_j & \frac{h}{6} & \frac{h}{3} & \frac{h}{3} & \frac{h}{6} \end{array}$	$\frac{h}{6} \to \langle u1 \rangle$	!Q18	$\frac{h}{6} \to \langle u1 \rangle$			
	$\frac{h}{6} = \langle u5 \rangle$	!Q19	$\frac{h}{6} \to \langle u5 \rangle$			
$[P_3(x_{ij}) + f(t_{ij})]\, q_j = r_{ij}$	$r_{ij} \to \langle z1 \rangle$	DA	$\langle z1 \rangle \cdot \langle u1 \rangle \to \langle z1 \rangle$	DB	DB	
$v_{i0} + \sum_{k=0}^{j} r_{ik} = g_{10}$	$g_{10} \to \langle p6 \rangle$	DB	$\langle z1 \rangle + \langle p6 \rangle \to \langle p6 \rangle$	EA	EA	
$v_{ij} \cdot q_j = s_{ij}$	$s_{ij} \to \langle z3 \rangle$	EA	$\langle p5 \rangle \cdot \langle u1 \rangle \to \langle z3 \rangle$	EB	EB	
$x_{i0} + \sum_{k=0}^{j} s_{ik} = g_{11}$	$g_{11} \to \langle p3 \rangle$	EB	$\langle z3 \rangle + \langle p3 \rangle \to \langle p3 \rangle$	EC	EC	
	$3 \to \langle u2 \rangle$	!Q20	$3 \to \langle u2 \rangle$			
	$3 \to \langle u6 \rangle$	!Q21	$3 \to \langle u6 \rangle$			
	$m_j s_{ij} \to \langle z3 \rangle$	EC	$\langle z3 \rangle \cdot \langle u2 \rangle \to \langle z3 \rangle$	ED	ED	
$x_{i,j+1} = x_{i0} + m_j s_{ij}$	$x_{i,j+1} \to \langle p2 \rangle$	ED	$\langle z3 \rangle + \langle p1 \rangle \to \langle p2 \rangle$	EF	EF	$TA \cdot 2$
	$m_j r_{ij} \to \langle z1 \rangle$	EF	$\langle z1 \rangle \cdot \langle u2 \rangle \to \langle z1 \rangle$	EG	EG	TE
$v_{i,j+1} = v_{i0} + m_j r_{ij}$	$v_{i,j+1} \to \langle p5 \rangle$	EG	$\langle z1 \rangle + \langle p4 \rangle \to \langle p5 \rangle$	FA	FA	ph j 0
	$2 \to \langle u7 \rangle$	!Q22	$2 \to \langle u7 \rangle$			$TE \cdot 2$
		FA	$B(FA) + \langle u7 \rangle \to FA$	TA	TA	TF
						phi j 1
						$TF \cdot 2$
						TG
						phi j 2
						$j + 1 \to j$

Fig.5.4 - Continuation of Fig.5.3

Analysis	Vocabulary	Index	Instruction	Operational Symbol	Remarks
$j+1 \to j$ $\quad 3:2 = \frac{3}{2}$	$\frac{3}{2}$	⟨n2⟩	TA	⟨n6⟩ : ⟨n7⟩ → n2	TH TH
$\frac{h}{6} \cdot 2 = \frac{h}{3}$	$\frac{h}{3}$	⟨n1⟩	TB	⟨n5⟩ . ⟨n7⟩ → n1	TC TC
$\frac{h}{6} \cdot 3 = \frac{h}{2}$	$\frac{h}{2}$	⟨z1⟩	TC	⟨n6⟩ . ⟨n5⟩ → ⟨z1	TD TD
$t_{i,j-1} + \frac{h}{2} = t_{ij}$ *jen pro* $j = 1, 3$	t_{ij}	⟨p7⟩	TD	⟨z1⟩ + ⟨p7⟩ → ⟨p7	AA AA
$m_2 = 3$	3	⟨n2⟩	TE	n6 + ⟨0⟩ → ⟨n2	BA BA
$q_2 = \frac{h}{6}$	$\frac{h}{6}$	⟨n1⟩	TF	n5⟩ + 0 → n1	TC TC
	$r_{i+1,0}$	⟨p4⟩	TG	H2⟨p6 + 0 → p4	TH TH
	$x_{i+1,0}$	⟨p1⟩	TH	H2 p3 + ⟨0⟩ → ⟨p1	TI TI
	$t_{i+1,0}$	⟨p7	TI	H2⟨p7 0 → p7	TJ TJ
	$r_{i+1,0}$	⟨p2	TJ	p6 + 0 → p5	TK TK
	$x_{i+1,0}$	⟨p5⟩	TK	⟨p1⟩ + 0⟩ → ⟨p2⟩	TL TL
	8	⟨n0	!Q23	8 → n0	
			TL	H: FA n0 → ! FA	TM TM
$i+1 \to i$	t_{max}	⟨n10⟩	!Q24	t_{max} → n10	
$t_{max} - t_{i0}$	g	⟨z1⟩	TM	n10⟩ . ⟨p7⟩ → ⟨z1	ST BA
			ST	0 . ⟨0 → 0⟩	

Fig.5.5 - Continuation of Fig.5.3

[redacted]

CHAPTER 6

PROCESSING OF PERFORATED CARDS

OPERATION WITH PERFORATED CARDS

In contrast to the automatic calculator, which processes numbers and instructions automatically, the machine employing perforated cards processws automatically only the numbers perforated in the perforated cards. The instructions must be set in the machine by the operator, i.e., a person must operate the machine.

6.1 Perforated Cards

For calculating purposes we distinguish between instruction cards, which contain as written symbols the instructions for the operator (Fig.6.1) and number cards

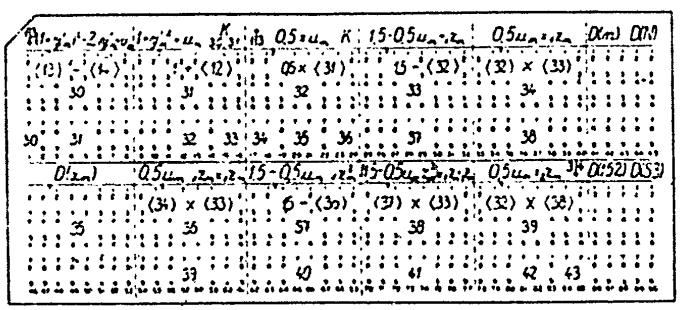


Fig.6.1 - Instruction Card for Set 3

on which the numbers entering into the operations are designated according to a particular code (Fig.6.2). For both the instruction card and the number cards, preprinted, ninety-column perforating cards are used. As a rule, standard cards are used, for operations with standard numbers. The mode of operation with standard cards does not differ from that with ordinary cards. The standard card is purposely divided by lines (designated by ruling in Figs.6.1 and 6.2) into ten number fields of eight columns each: 1 to 8, 9 to 16, ..., 78 to 85. Columns 41 to 45 and 86

to 90 are reserved for indexing the individual cards, for correlating the set of cards, and for perforation of the simole constants which during the calculation come up repeatedly or very frequently. Every field of the standard card has its serial number, and can be filled with a standard number, which is composed of the sign and 7 digits.

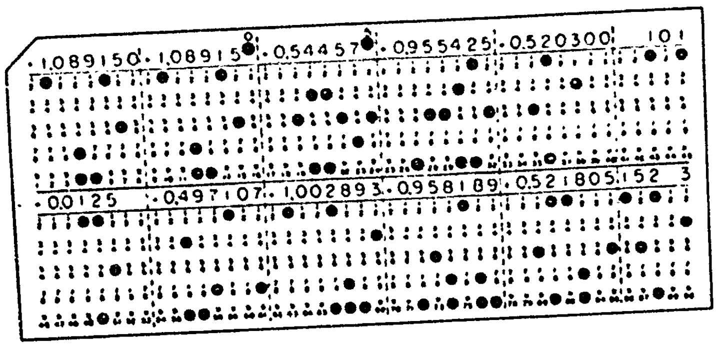


Fig.6.2 - Number Card of Index 101 (Columns 43 to 45), Set 3 (90 Columns). With the card were carried out operations 31 to 42 according to the instruction caard (Fig.6.1). (The title numbers are actually printed on the card)

The instruction card has number fields distinguished by serial numbers. An example of an instruction card is shown in Fig.6.1. On the first line of each of the fields of the card (on the unprinted part) the mathematical statement of the operation is written whose result is to be perforated in the field for this operation. (The symbol D ( ) means that the value must be put in parentheses before beginning the operation of perforating the corresponding number card). The second line states the same operation with the help of the serial numbers of the fields in which the values are present which enter into the operation. On the third line is the serial number of the field (for example, 30 to 39) and on the fourth line the serial number of the operation (for example, 30 to 43). A decimal point

is placed in each of the designated fields.

The symbol $^{1}{\downarrow}3$ in the first row of Fig.6.1 means: "Arrange the set of cards 1 relatively to set of cards 3 in such a way that the nth card of set 3 follows the nth card of set 1!" The symbol $_{30}K_{31}$ means: "Verify the agreement between the content of field 30 and the content of field 31, and the correctness of the perforation at the place designated by the letter K!" The symbol K means: "Verify the previous operation and the correctness of the perforation at the place designated by the letter K!" The symbol $_1\uparrow_3$ means: "Separate set 1 from set 3!" D(n) means: "Perforate before beginning to calculate with the cards of the corresponding Index n!". Similarly D(N), D(x_n), D(152) and D(S3) means: "Perforate the number N, the value x_n the constants 1, 5, 2 and the numbers of set 3!"

The standard number card with ten number fields (Fig.6.2) corresponds to one row of the ten-column calculating form (see Paragraph 1.2). Each of the fields contains the sign in its first column. If the sign is positive it is without a perforation in the first column; if it is negative, there is a perforation in the first column at the place preprinted with $\frac{3}{4}$. In the second to the eighth columns each of the fields is reserved for desitnating a seven-place number. The odd digits are designated by a perforation in the corresponding column, in some places by a preprinted pair of numbers. The upper digit of the preprinted pair is read (for example, the perforation in the place of the preprinted pair $\frac{7}{8}$ denotes the digit 7). Even digits are denoted by two perforations in the corresponding column. One of these perforations is always together with a preprinted nine. In reading, the lower digit of the number pair is valid at the place where the upper one is perforated. Figure 6.2 shows an example of a number card with the titles given. The numbers 1 to 90, which are printed below the lines denote the numbers of the columns. A decimal point defines the decimal place.

An instruction card corresponds to an instruction row of the calculating form. Accordingly, the individual number cards correspond to the individual number rows

STAT

of the form, and the number fields of the set of cards which have the same serial number correspond to the columns of the form. Every number card is designated by an index (in Fig.6.2, for example, in columns 41 to 45), which corresponds to the serial number of the corresponding row of the form. A pack of cards corresponding to all of the row of the first to the tenth columns of the calculating form is called a set. Since a row of the calculating form usually has more than 10 columns, it is necessary to "lengthen" the cards of set 1 with an additional card of set 2, and, if necessary, with one of set 3, etc. (in Fig.6.2 the numbers of sets are designated in columns 89 and 90).

6.2 Operation with Cards and Operation with Numbers

To each set of number cards belongs one instruction card.

According to the data on the instruction cards, the operation with the cards and the operation with the numbers are carried out. The operation with the cards is carried out with the classifier, and the operation with the numbers with the perforator, the calculating perforator and the tabulator.

In the operation with the cards, combination and classification of the number cards are employed. The operation of combination is used, for example, when one set of number cards is combined with the following set in such a way that the first card of the second set follows the first card of the first set, the second card of the second set follows the second card of the first set, etc. This combination is carried out with the classifier, for example, according to the perforated indices in the index fields. In the arrangement of the cards, the pack of cards formed by the combination of two sets, for example, must be reclassified into the first and the second set. Also when tabulated values of functions have to be employed in the calculation, the "tabulation" of the cards is carried out according to the argument of the corresponding set of cards, and the corresponding functional values are calculated by interpolation*. (For footnote, see next page)

With the perforator, the initial values and the indices of the individual cards are perforated into the cards.

With the calculating perforator the following operations are carried out:

a) Operations on the same cards (Fig.6.3)

aa) Multiplication of two numbers perforated in any two fields of the same card and perforation of the result in any empty field of the same set.

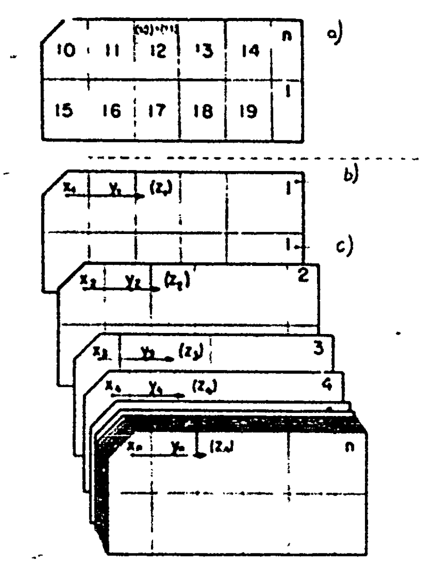


Fig.6.3 - Operation on the Same Card.

$x_i \times y_i = z_i$

a) Instruction card with instruction for field 12; b) Card No. (Index); c) Set No.

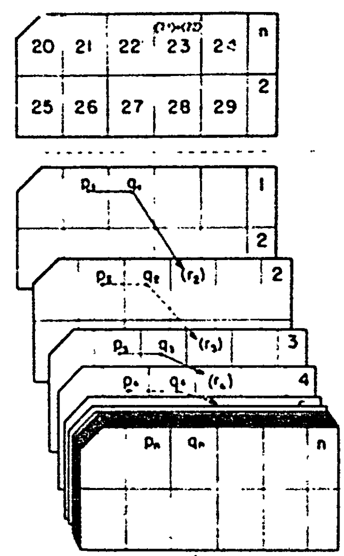


Fig.6.4 - Operation from Card to Card.

$p_i + q_i = r_{i+1}$ (The broken lines denote operations in the second "run" of the machine)

---

* It must be borne in mind that this operation must be carried out by machine because several hundred cards are always processed at the same time

ab) Addition of two numbers perforated in any two fields of the same card, and perforation of the result in any empty field of the same card.

ac) Raising to the second power the number perforated in any field, and perforation of the result in any empty field of the same card.

ad) Division of two numbers perforated in any two fields of the same card, and perforation of the result.

b) Operations with card to card

ba) Multiplication of two numbers perforated in any two fields of an odd card, and perforation of the result in the next even card.

bb) Addition of two numbers perforated in any two fields of an odd card, and perforaation of the result in the next even card.

bc) Raising to the second power the number perforated in any field of an odd card, and perforating the result in the next even card.

For carrying out any one of operations b), one empty card is placed in front of the pack. In this way an odd card is always next to an even card. At the next passage of the pack through the machine, the results are also obtained on the originally even cards.

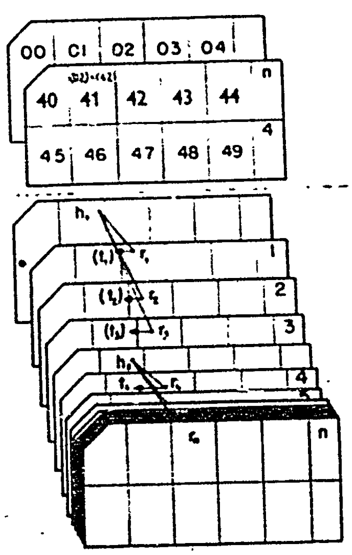


Fig.6.5 – Operation from Leading Card to Trailing Cards. $h_1 \times r_j = t_j$

c) Operations from leading Card to Trailing Card (Fig.6.5)

The leading card is denoted by the perforation present in the left edge between the upper and lower half of the card (see Figs.6.6 and 6.7). The value perforated in this card is retained in the

STAT

machine until no leading cards are any longer present.

ca) Multiplication of the number perforated in the leading card by the numbers perforated in the trailing cards (until the next leading card), and perforation of the results in the card containing the second factor.

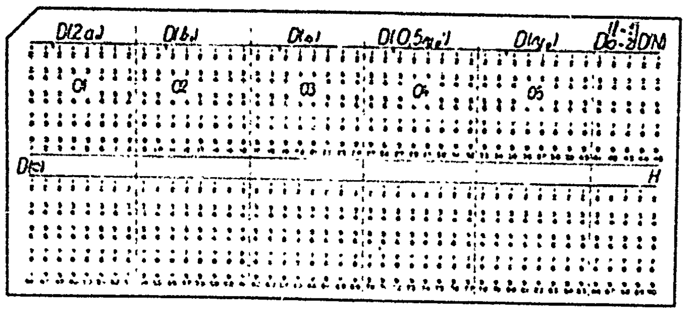


Fig.6.6 - Instruction Card for Perforation of Leading Card by Perforator

cb) Addition of the number perforated into the leading card with the numbers perforated into the trailing cards (until the next leading card), and perforation of the results into the card containing the second term.

d) Operations from Leading Card to Trailing Even Cards [with Leading Card Followed by Empty Card (Fig.6.8)]

da) Multiplication of number in leading card by number in odd card, and perforation of result into even card.

db) Addition of number in leading card with number in odd card, and perforation of result into even card.

e) Operations along Pack of Cards (Fig.6.9)

ea) Successive subtraction of values perforated into the same field of all cards of one set, and perforation of the intermediate results into the individual cards.

135

eb) Successive addition of values perforated into the same field of all cards of one set, the intermediate result taken with the opposite sign, and perforation

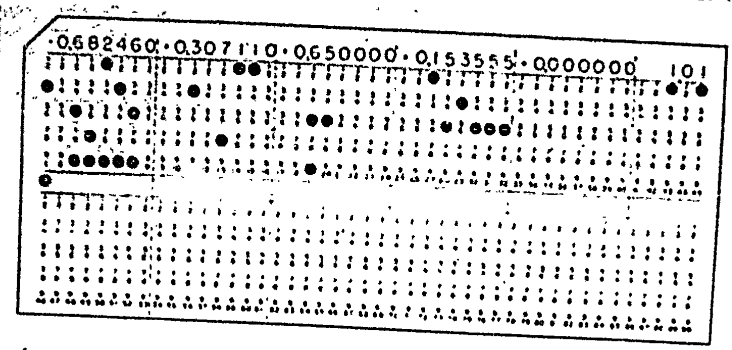
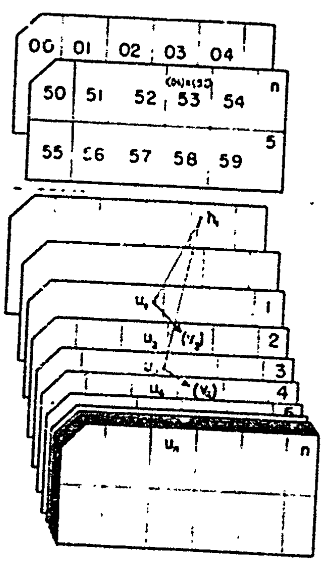


Fig.6.7 - Leading Number Card with Perforated Values (the Title Numbers are not Actually Printed on the Card)



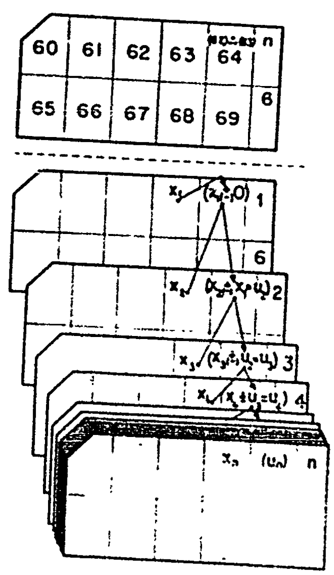


Fig.6.8 - Operation from Leading Card to Trailing Even Cards. $h_i \times u_j = v_j + 1$

Fig.6.9 - Operation Along Pack of Cards. Successive subtraction (addition) of value x along set 6

of the intermediate results into the individual cards.

All of the enumberated operations occur automatically with respect to the sign.

The operation of the tabulator is usually combined with the printing of the values perforated into the cards and the subtraction of the values perforated into the particular columns of the card (at the processing of the cards with the tabulator, the card columns are not divided into individual fields) with simultaneous printing of the sums, the intermediate results, and the grand totals.

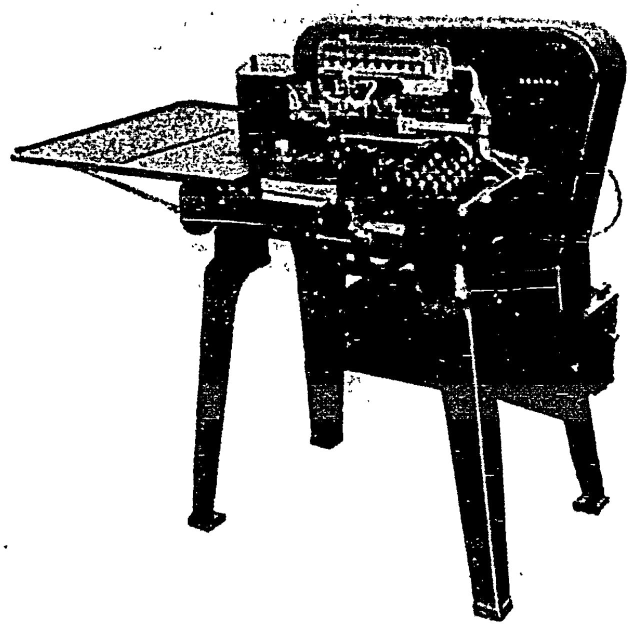


Fig.6.10 – Perforator

6.3 Perforator

The perforator is an electrically-driven machine, which perforates the initial values, the index, the constants, the perforation of the leading card, etc. into the perforating cards. The pack of cards to be perforated with the information according to the written or printed data is placed into the feed magazine with the

face of the cards upward and the cut corner to the left. By pressing the number keys of the keyboard, the necessary digits are set in the corresponding columns 1 to 90. In contrast to the typewriter, where the roll with the paper moves during the typing of the letters and digits and the writing is obtained directly by pressing the character keys, the setting carriage of the perforator moves from the first to the 90th column, giving only the setting. When the "operating key" is pressed

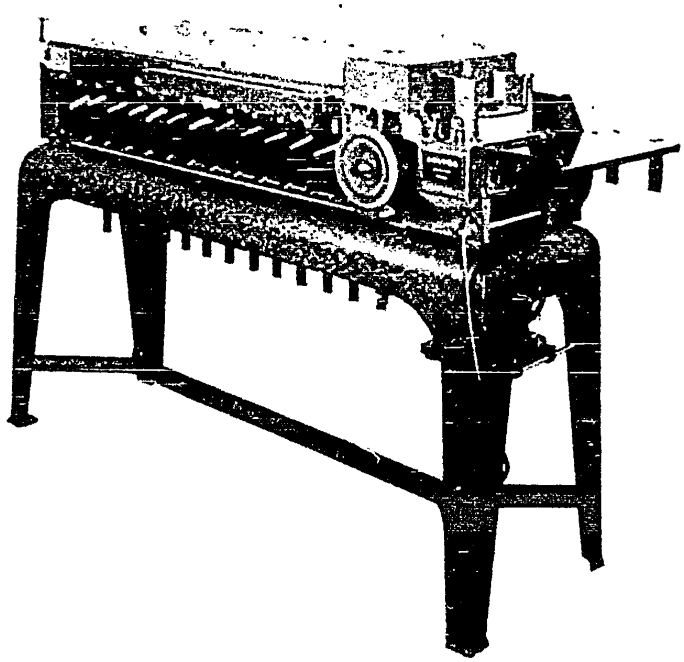


Fig.6.11 - Classifier

all of the set data are perforated into the card, the card moves automatically to the storage magazine, and the machine feeds the next lowest card. The set values remain intact, and the machine perforates the cards at a rate of 5000 to 6000 cards per hour, or the set values are automatically cancelled and the operator sets the new values for the next cards.

Verification of the correctness of the perforating is obtained by repeating the perforating, the perforation slightly displaced (by half the perforation). This verified perforated card then contains only oval holes which can easily be verified, even by eyesight.

6.4 Classifier

The classifier is used for carrying out operations with the cards. The pack of cards is placed in the feed magazine in the same way as in the perforator. The feeder feeds the cards of the pack one by one from below to the machine at a rate of 24,000 cards per hour. Each card stops for a moment under the feeler of the carriage, where the column is determined according to which the card is to be classified, and where the carriage will stop. According to the information contained in the determined column, one of the thirteen* storage compartments opens. (Ten of these thirteen compartments correspond to the ten digits 0 to 9). The transporting roll moves the card to the open compartment into which it falls.

In this way, by one "pass" through the machine the pack of cards can be resorted into as many batches as there are different kinds of information in one column. If, for example, a pack is to be reclassified consisting of two kinds of sets, for example, set 14 and set 15 (whose number indices are designated in columns 89 and 90), then the feeler of the carriage is set for column 90. The cards with the digit 4 perforated into column 90 falls into compartment 4, while the cards perforated with the digit 5 fall into compartment 5.

The combining of cards is carried out in such a way that the pack is first sorted according to the lowest index. The cards are then again made up into a pack in such a way that the lower card is taken from compartment 0, then from compartment 1, then from compartment 2, etc., until finally from compartment 9.

---

* The classifier is, of course, also used for processing other types of cards, which, however, is of no interest to out present purpose

STAT

Then the pack is classified according to the next higher index. Since the machine feeds the cards of the pack from below, first the cards the the mark 0 fall into the compartments 0 to 9, then the cards with the mark 1, etc. in accordance with the second order given in the corresponding column. The batches obtained in this way are again put together in such a way that the lower card is fed from compartment 0, then from compartment 1, etc. This pack is again classified according to

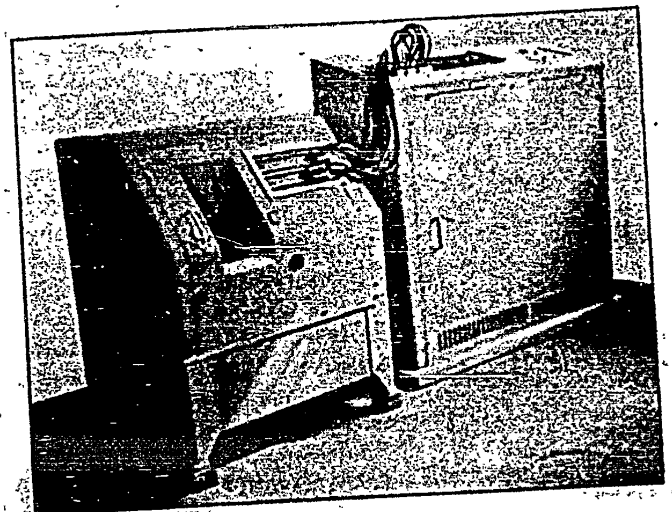


Fig.6.12 - Calculating Perforator

the columns with successively higher indices to the highest, and by repeated combination we have obtained the card arrangement according to the indices. If the index is composed of several different marks, the combining is carried out in the same manner. First, however, it must be decided to which series the individual marks correspond.

6.5 Calculating Perforator

The calculating perforator is a semiautomatic relating calculator which automatically processes according to the present perforating operation a perforating datum in a perforating card and perforates the result into the same or into the

STAT

191

next card. The operation, which is described above (Paragraph 6.2), is set by hand on the control board of the machine. The field to be perforated is set on the perforating part of the movable perforator carriage.

The pack of cards is placed in the feed magazine in the same way as with the perforator or the classifier. The machine is fed from below one by one card at the rate of 6000 or 3000 cards per hour. The fed card stops for a moment under one of the feelers where it is felt. An electric signal corresponding to the datum on the card is transmitted to the control baord. From here the signal is transmitted only from the number field participating in the operation to the arithmetical unit. In a split second the arithmetical unit carries out the operation and sends the result via the control board to the setting electromagnet of the perforating part of the machine. Here the card is fed to the perforating part of the machine and again stops. In the next moment the card is perforated with the datum set by the electromagnet on the perforator carriage. The perforated card is then placed in the file.

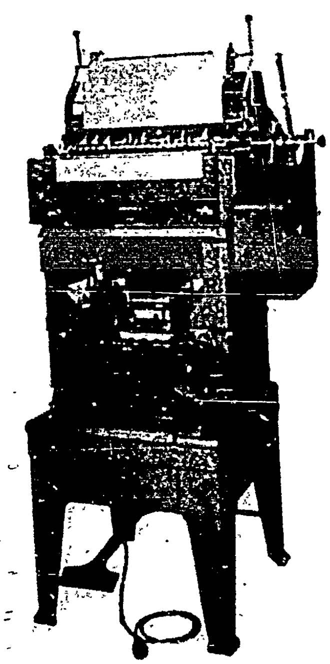


Fig.6.13 - Tabulator

The machine processes numbers with regard to the sign. In the use of standard numbers it carries out 6000 multiplications or additions of two 7-place numbers or 3000 divisions of two 7-place numbers per hour. Operating with single cards in succession, the rate of the machine is 3000 operations per hour.

243

The operation of feeling the card carries out the calculation and the perforation of the result. Every operation is verified by the verifying operation while processing the next card in the calculating perforator. Multiplication is verified by exchanging the factors and by changing the order of one of the factors. This assures that the other parts of the machine participate in the verifying calculation. The probability is minimal that the same error will occur in both the original and the verifying calculation in exactly the same two different places of the machine. The result of the verifying calculation is not perforated in the card. The card with the verified result is, however, felt, and compared in the machine. If the two results are in agreement, the machine perforates into the card above the verified result the words "perforation correct". The machine carries out this verifying operation likewise at a rate of 6000 operations per hour.

The verification of addition is carried out in such a way that in the second calculation the terms are added with the opposite signs. The machine therefore operates with both terms filled up with nines with entirely different digits in the same decimal place. Also in this case the probability is very small that an error in the machine will occur at the same decimal place of the corrected digits. The verified result is, however, perforated in some empty field of the card. By another calculation the verified result is added with the result to be verified. If the addition comes out zero, the machine perforates the words "perforation correct" above the verified result. The verification of addition takes two normal operations. It is therefore carried out at a rate of 3000 operations per hour.

Division is verified by multiplying the quotient by the divisor without perforating and by comparing the product with the divident for correctness of the perforation.

6.6 Tabulator

In the solution of mathematical problems, the tabulator is usually used for

STAT

033

printing the results. The pack of cards is placed into the feed magazine in the same way as with the other described machines. The machine is fed from below with card after card at a rate of 6000 cards per hour. Every card stops for a moment in the feeler chamber, where it is felt by the feeler needle. According to the felt information, the machine sets the corresponding characters of the writing segment against the writing roll, and by touching the typing keyboard the felt information is printed on the paper roll by the writing ribbon.

The tabulator is also provided with a subtracting position and a subtraction column for the purpose of entry. In a reversed movement of the gear of the writing segment to the starting position, this segment engages with the gear of the subtractor, and, depending on the sign, the present and printed value is either added or subtracted. On the signal "print the intermediate result", the segment engaged with the subtractor is already moving forward (setting), and its end position is determined by the content of the subtractor. Here the setting segment cancels the content of the subtractor. After the intermediate has been printed, the segment, in forward movement, is in engagement with the gear of the main subtractor instead of with the gear of the nullified subtractor, so that the already-printed intermediate result is added to it. On the signal "print the main sum total", the writing segment in its forward movement is in engagement with the gear of the totalling subtractor, which sets its content into the terminal position of the writing segment. After the grand total has been printed, this subtractor is cancelled.

[redacted]

STAT

CHAPTER 7

EXAMPLE OF SOLVING A TECHNICAL PROBLEM BY MACHINES FOR THE PROCESSING OF PERFORATED CARDS

PROBLEM AND GIVEN VALUES

7.1 Coordinate Table for the Production of Compressor Blades

Only in large-scale production of compressor blades is the copying of a model on a coordinated milling machine economical. For this purpose it is necessary to calculate the coordinates of a large number of points of a curve equidistant to the periphery of the profile section of the blades. The equidistant clearance from the periphery is obviously equal to the radius of the working tool. The coordinate system is first selected in such a way that the beginning falls in the "entrance corner" of the profile and the X-axis passes through the "exit corner". Also the depth of the profile c, which is equal to the distance of the exit corner from the entrance corner, is at first selected equal to 1. It is only at the end of the calculation that the reduction of the coordinates to the required depth of the profile and the transformation of the coordinates to the given coordinate system are carried out.

Let us discuss the problem of calculating for each of 33 profiles 110 points equidistant to the top of the periphery and 110 points equidistant to the bottom of the periphery, i.e., a total of 7260 points. Parallel calculation of such a number of values can be carried out advantageously with the machine for processing perforated cards. Since the principal part of the work is done on the calculating perforator, which, during the processing of the values, perforates in the same card or in adjacent cards, a very-restricted interpolation method is selected. The requirement that the interpolation curve must be "smooth and not "undulating in the limits of the precision makes us formulate a method which is simple with respect to the employment of the machines and at the same time satisfies the required

precision.

7.2 Bases for the Solution of the Problem

For each profile N ($N = 1, 2, \ldots, 33$) are given the numerical values m_N and p_N for calculating the center line [the coordinates of 17 points x_n, d_n^* situated symmetrical to the profile ($n = 0, 1, \ldots, 16$) for $x_n = 0$; 0.0125; 0.025; 0.05; 0.075; 0.1; 0.15; 0.2; 0.3; ..., 0.8; 0.9; 0.95; 1.00 and for the depth of the profile $c^* = 1$] tabulation of the thicknesses symetrical to the profile t_t and the required thickness of the profile t_N, the required thickness of the profile c_N, the radius of the tool r and the position of the profile in the given coordinate system coordinating the "entrance corner" (X_o, Y_o) and the angle γ, which includes the joining line of the "entrance corner" and the "exit corner" with the X-axis.

The center line is composed of two parabolic curves given by the equations

$$y = \frac{m}{p^2}(2px - x^2) \qquad \text{for} \qquad 0 \leq x \leq p \tag{7.1}$$

$$y = \frac{m}{(1-p)^2}[(1-2p) + 2px - x^2] \qquad \text{for} \qquad p \leq x \leq 1 \tag{7.2}$$

Calculated first are the ordinates of 17 points of the center line for the thickness of profile $c^* = 1$ at putting $x = x_n$ in eqs.(7.1) and (7.2), where the value x_n agrees with the value of the abscissa of the symmetrical profile. In these 17 points of the center line, on the normal to the center line, laid out in both directions is the distance d_n ($n = 0.1, \ldots, 16$), which are the ordinates d_n^* of the fundamental symmetric profile of thickness t_t, reduced to the required thickness t_N, so that $d_n = d_n^* t_N / t_t$.

By interpolation between the 17 points of each of the peripheries, always 110 points (x_i, y_i) are obtained for $x_i = 0.005$; 0.01; 0.015; ...; 0.095; 0.01; 0.02; ...; 0.98; 1.00; 0.02; ...; 0.98; 0.99; 1.00. Thus the first derivation must be continuous and the second derivation must be nonoscillating. The required

precision supposedly is 0.001 of the thickness of profile.

After carrying out the reduction of the coordinates with respect to the required thickness of profile c, the corresponding points of the equidistant curve laid out at a distance equal to the radius of the working tool on the normal to the interpolation curve are calculated for 110 points of the periphery. Finally,

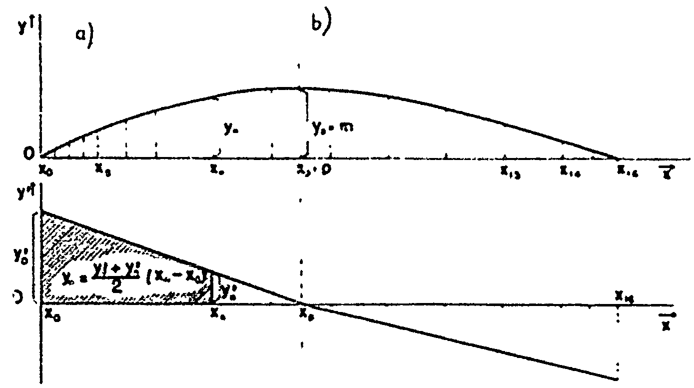


Fig.7.1 - Center Line of the Profile and its First Derivation

a) Left curve; b) Right curve

the transformation of the coordinates to equidistance in the coordinate system is carried out.

Solution

First, we select a method suitable for the solution of the problem on the basis of our case, using the perforated-card processing machine. The individual operations are written on the instruction cards (see Fig.6.1). According to the instruction cards, we prepare a legible operation table with indications and explanations for serving the machine and indications for perforating the initial values, indices, and constants.

7.3 Preparation for the Calculation of the Fundamental Values, the Center Lines

For the calculation we start with the equation of the parabola

$$y = ax^2 + bx + c \qquad (7.3)$$

and with its first derivation

$$y' = 2ax + b\,. \qquad (7.4)$$

First we calculate the y'_n values corresponding to the abscissa x_n($n = 0.1, \ldots, 16$)

$$y'_n = 2ax_n + b \qquad (7.5)$$

and integrate the corresponding y_n values according to the relation

$$y_n = y_0 + \tfrac{1}{2}(y'_0 + y'_n)(x_n - x_0)\,. \qquad (7.6)$$

The center line is composed of two parabolic curves (Fig.7.1): the left curve for $0 \leq x \leq p$, and the right curve for $p \leq x \leq 1$. By comparing eqs.(7.1) and (7.2) with eq.(7.3), we obtain for the coefficients of the left curve the values

$$a_1 = -\frac{m}{p^2}\,; \qquad b_1 = \frac{2m}{p}\,; \qquad c_1 = 0\,;$$

Valid (see Fig.7.1) are additionally

$$x_0 = 0\,; \qquad y_0 = 0\,; \qquad y'_0 = b_1 = \frac{2m}{p}\,.$$

For the coefficients of the right curve we obtain the values

$$a_2 = -\frac{m}{(1-p)^2}\,; \qquad b_2 = \frac{2pm}{(1-p)^2}\,; \qquad \left(c_2 = \frac{(1-2p)\,m}{(1-p)^2}\right);$$
$$x_0 = p = x_p\,; \quad y_0 = m = y_p\,; \quad y'_0 = 0 = y'_p\,.$$

For each profile N we perforate one main card H_1 with the values $2a_1$ and b_1 necessary for the calculation of the y'_n of the left curve, and the values y_n (see Figs.6.6 and 6.7). Further we perforate for each profile another main card H_2

with the values $2a_2$, b_2, $\overline{y}_o = m = y_p$ and $\overline{y}_o' = 0$ which are necessary for the calculation of the values of the right curve. At the same time we perforate into both main cards of each profile the value $s_N = t_N/t_t$ necessary for calculating the fundamental points of the periphery. In this way, we prepare 33 main cards H_1 for 33 profiles for calculating the left curve and 33 main cards H_2 for calculating the right curve, which we then file as a set of cards S1. Always 17 cards are denoted with the same index N corresponding to the number of the profile ($N = 1.2, ..., 33$), and are always placed together in a set. For each profile 17 cards are perforated with the value x_n as well as the ordinate d_n^* of the fundamental symmetrical profile necessary for calculating the fundamental points of the periphery ($n = 0.1, ..., 16$). Simultaneously we prepare a set of cards S2, which is also composed of 33×17 perforated cards. Perforated into these cards besides the number of the profile and the value x_n is the value $(x_n - x_o)$, in the cards with the value x_n is valid $0 \leq x_n < p$, consequently, $x_n = 0$, and in the cards with the value x_n with $p \leq x_n \leq 1$ we have $x_o = p = x_p$.

The main card H_1 for the left curve is placed before the first card of the individual profile S1 or S2. The main card H_2 for the right curve is placed before that card of the corresponding card into which is perforated the minimum value x_n at $x_n \leq p$. The corresponding operating table 1 is shown in Fig.7.2.

Explanations to Table 1: Into the first column, with the heading "Mathematical Statement", are entered the magnitudes perforated into the perforated cards before the beginning of the operation, and the mathematical expressions of the operations. Into the second column, entitled "Set", the sets are entered which participate in the corresponding operations. H denotes a set of main cards (in our case 33 cards H_1 for the left curve and 33 cards H_2 for the right curve), S1 set 1, S2 set 2, etc. In the third column, "Operation No.", is entered besides the serial number of the operation also the letter D, which means that the value present on this line of the first column is perforated into the perforated card of the corresponding set before

STAT

46

Problem: Calculating the Equidistant Coordinates of 33 Profiles

Table 1

Page 1

1. Center Line

Mathematical Statement	Set	Op. No.	Operation Symbol: Operations	Operation Symbol: where	Remarks
$2a$	H	D		01	
b		D		02	$s = \frac{t_x}{t_t}$
s		D		03	
$0.5y'_0$		D		04	
y_0		D		05	
x_n	$S1$	D		15	
d^*_n		D		16	
		1	$H \downarrow S1$		
$2a \times x_n = 2ax_n$	H	2	⟨01⟩×⟨15⟩	17	
		3	K(⟨15⟩×⟨01⟩)	K 17	
$b \div 2ax_n = y'_n$		4	⟨02⟩ − ⟨17⟩	18	
		5	⟨02⟩ ⟨17⟩	19	
		6	K(⟨18⟩, ⟨19⟩)	K 18	
$s \times d^*_n = d_n$		7	⟨03⟩×⟨16⟩	10	
		8	K(⟨17⟩×⟨03⟩)	K 10	
		9	$H \uparrow$		
x_n	$S2$	D		25	
$(x_n - x_0)$		D		26	
		10	$S1 \downarrow S2$		
$0.5 \times y'_n = 0.5y'_n$	$S1$	11	0.5×⟨18⟩	20	
		12	K(⟨18⟩×0.5)	K 20	
		13	$S1 \uparrow$		
		14	$H \downarrow S2$		
$0.5y'_0 + 0.5y'_n = (0.5y'_0 + 0.5y'_n)$	H	15	⟨04⟩ ÷ ⟨20⟩	21	
		16	−⟨04⟩ −⟨14⟩	22	
		17	K(⟨21⟩, ⟨22⟩)	K 21	
$y_0 \times 1 = y_0$		18	⟨05⟩×1	23	
		19	K(1×⟨05⟩)	K 23	
		20	$H \uparrow$		
$(0.5y'_0 + 0.5y'_n) \times (x_n - x_0) = ( ) \cdot ( )$		21	⟨21⟩×⟨26⟩	24	
		22	K(⟨26⟩×⟨21⟩)	K 24	
$(0.5y'_0 + 0.5y'_n)(x_n - x_0) + y_0 = y_n$		23	⟨24⟩ ÷ ⟨23⟩	27	
		24	−⟨24⟩ − ⟨23⟩	28	
		25	K(⟨27⟩, ⟨28⟩)	K 27	

Fig. 7.2 – Operating Table for Calculating the Center Line

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

the beginning of the operation. The next two columns, entitled "Operation Symbol", contain in the first part the symbolic expression of the corresponding operation. H | S1 means: "place main card (of set H) into set 1!" H | means: "take out main cards!" S1 | S2 means: "combine set 1 with set 2 in such a way that the nth cards of set 2 follow the nth cards of set 1" S1 | means: "take out set 1!" <01> × <15> means: "multiply content of field 01 (on main card) by content of field 15 (on next card)!" K<15> × <01> means: "verify the multiplication with the exchanged factors!" <02> + <17> means: "add the content of field 02 to the content of field 17!" <02> − <17> means: "add the contents with the opposite signs!" K<18>, <19> means: "verify agreement of content of field 18 with content (having the opposite sign) of field 19!" In the second part of the column, entitled "where", is entered the number of the field in which the result of the operation is to be perforated. The symbol K17 means that the result is not perforated but that only "perforation correct" is perforated above field 17.

7.4 Calculation of Fundamental Points of the Periphery

In 17 points (x_n, y_n) of the center the normals are erected. On these normals, the distance $d_n = d_n^* s$ is laid out, where d_n^* is the ordinate of the symmetrical profile of tabulated thickness t_t and $s = t_N/t_t$. The coordinates of the symmetrical profile x_n and d_N^* are given in the Table with the given tabulated thickness of the profile t_t. The required thickness of the profile t_t is a particular datum for each profile N. Figure 7.3b shows the symmetrical profile of reduced thickness t_N. From Fig.7.3a it is known that

$$y'_n = \operatorname{tg} \gamma = -\frac{\delta x_n}{\delta y_n} \tag{7.6}$$

and also that

$$(\delta x_n)^2 + (\delta y_n)^2 = d_n^2 \tag{7.8}$$

In eq.(7.8) we take for δx_n from eq.(7.7) $\delta x_n = -y'_n \delta y_n$. Thus we get

STAT

$$(\delta y_n)^2 y_n'^2 + (\delta y_n)^2 = d_n^2 ,$$

whence

$$\delta y_n = d_n(1 + y_n'^2)^{-\frac{1}{2}} . \qquad (7.9)$$

We put

$$1 + y_n'^2 = u_n \text{ and } u_n^{-\frac{1}{2}} = z_n .$$

Using the iteration equation for calculating $z = u^{-\frac{1}{2}}$ gives

$$z_{m+1} = z_m(2 - u_n z_m^2 + 1)/2 = z_m(1.5 - 0.5 u_n z_m^2) \qquad (7.10)$$

where the index m belongs to the result after the nth iteration step. The number

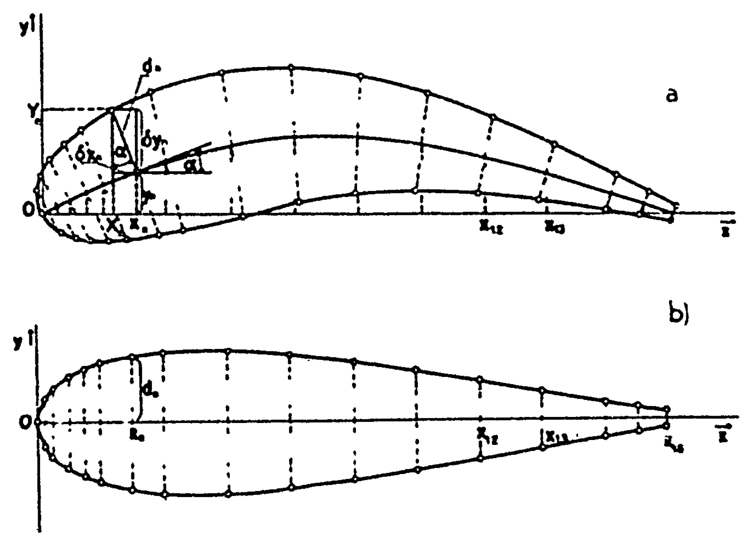


Fig.7.3 - a) Plotting of ordinates of symmetrical profile reduced to center line; b) Symmetrical profile reduced to thickness t_N; c) Upper periphery; d) Lower periphery

STAT

of iteration steps required for the calculation is determined in such a way that

$$ {}_{m+1}z - {}_{m}z < \varepsilon , $$

where ε is selected according to the desired precision of the calculation.

The coordinates of the points of the upper periphery of the profile are given by the relations

$$ X_n^A = x_n - \delta x_n , \quad Y_n^A = y_n + \delta y_n , \quad (n = 0, 1, \ldots, 16) . \tag{7.11} $$

and the coordinated of the points of the lower periphery of the profile by

$$ X_n^d = x_n + \delta x_n , \quad Y_n^d = y_n - \delta y_n , \quad (n = 0, 1, \ldots, 16) . \tag{7.12} $$

consequently,

$$ \delta y_n = d_n \cdot {}_{m}z_n , \quad \delta x_n = -y'_n \cdot \delta y_n , \quad (n = 0, 1, \ldots, 16) . \tag{7.13} $$

The operations corresponding to Pages 1, 2, 3 of Table 2 are presented in Figs.7.4, 7.5, 7.6.

Explanations to Table 2: The calculation of $1 + y'^2_n = u_n$ is carried out with operations 26 to 33. The verification of the raised power is carried out by calculating $1 + y'^2_n$ by two different methods and the two results are as follows:

$$ u_n = 1 + (y'_n \times y'_n) ; \qquad u_n = (1 + y'_n) \times (1 + y'_n) - 2y'_n . $$

In the calculation of $(1 + y'^2_n)^{-1} = u^{-1} = z$ we take as the first approximation ${}_0z_n = 1$. Inserting this into eq.(7.10), the first iteration step reduces to

$$ {}_1z_n = 1{,}5 - 0{,}5u_n \text{ (Operation 37, eq. 7.4)} $$

Already after carrying out the second iteration step we obtain in out case (where $y'_n < 0.2$) a result accurate to 5 decimals. The third iteration step is carried out for a comparing result. The fourth iteration step is planned for the event that the machine makes an error in calculation and that ${}_3z_n - {}_2z_n > 10^{-5}$. This fourth

STAT

347

Problem: Calculation of Equidistant Coordinates of 33 Profiles — Table 2

2. Calculation of Fundamental Points of Periphery — Page 1

Mathematical Statement	Set	Op. No.	Operation Symbol: Operations	Operation Symbol: where	Remarks
$1 - q'_n = (1 - q'_n)$	S1	26	1 − 18	11	
$q'_n \cdot q'_n = q'^2_n$		27	18 · 18	12	
$(1 - q'_n) \cdot (1 - q'_n) = (1 - q'_n)^2$		28	11 · 11	13	
$q'_n + q'_n = 2q'_n$		29	18 + 18	14	
	S3	D		45	
s_n		30	S1, S3		
$(1 - q'_n)^2 : 2q'_n = u_n$	S1	31	13 : 14	30	
$1 - q'^2_n = u_n$		32	1 − 12	31	
		33	K(30 · 31)	K 31	
		34	S1*		
$0.5 \cdot u_n = 0.5u_n$		35	0.5 · 31	32	
		36	K(31 · 0.5)	K 32	
$1.5 - 0.5u_n = {}_1z_n$		37	1.5 − 32	33	1. Iteration Step
$0.5u_n \cdot {}_1z_n = 0.5u_n \cdot {}_1z_n$		38	32 · 33	34	2. It. Step
$0.5u_n \cdot {}_1z_n \cdot {}_1z_n = 0.5u_n \cdot {}_1z_n^2$		39	34 · 33	36	
$1.5 - 0.5u_n \cdot {}_1z_n^2 = (1.5 - 0.5u_n \cdot {}_1z_n^2)$		40	1.5 − 36	37	
$(1.5 - 0.5u_n \cdot {}_1z_n^2) \times {}_1z_n = {}_2z_n$		41	37 · 33	38	
$0.5u_n \cdot {}_2z_n = 0.5u_n \cdot {}_2z_n$		42	32 · 38	39	
z_n	S4	D		45	
		43	S3, S4		
$0.5u_n \times 1 = 0.5u_n$	S3	44	32 × 1	40	... It. Step
		45	K(1 · 32)	K 40	
${}_2z_n \cdot 1 = {}_2z_n$		46	38 × 1	41	
		47	K(1 · 38)	K 41	
$0.5u_n \cdot {}_2z_n \cdot {}_2z_n = 0.5u_n \cdot {}_2z_n^2$		48	39 · 38	42	
		49	S3*		
$1.5 - 0.5u_n \cdot {}_2z_n^2 = (1.5 - 0.5u_n \cdot {}_2z_n^2)$		50	1.5 − 42	43	
$(1.5 - 0.5u_n \cdot {}_2z_n^2) \ldots$		51	43 · 41	44	
$0.5u_n \cdot {}_3z_n = 0.5u_n \cdot {}_3z_n$		52	40 · 44	46	4. It. Step
$0.5u_n \cdot {}_3z_n \cdot {}_3z_n = 0.5u_n \cdot {}_3z_n^2$		53	46 · 44	47	
$1.5 - 0.5u_n \cdot {}_3z_n^2 = (1.5 - 0.5u_n \cdot {}_3z_n^2)$		54	1.5 − 47	48	
$(1.5 - 0.5u_n \cdot {}_3z_n^2) \times {}_3z_n = {}_4z_n$		55	48 · 44	49	
		56	K(44 · 49)	K 49	

Fig. 7.4 – Operation Table for Calculating Fundamental Points of Periphery, p. 1

STAT

Problem: Calculation of Coordinates of 33 Profiles
2. Calculation of Fundamental Points of Periphery

Table 2
Page 2

Mathematical Statement	Set	Op. No.	Operation Symbol: Operation	Operation Symbol: where	Remarks
x_n	$S5$	D		53	
		57	$S2 \downarrow S5$		
$y_n \times 1 \to y_n$	$S2$	58	⟨27⟩×1	50	
		59	K(1×⟨27⟩)	K 50	
		60	$S2 \uparrow$		
		61	$S1 \downarrow S5$		
$d_n \times 1 \to d_n$	$S1$	62	⟨10⟩×1	51	
		63	K(1×⟨10⟩)	K 51	
$d_n \times y'_n = d_n y'_n$		64	⟨40⟩×⟨18⟩	52	
		65	K(⟨18⟩×⟨10⟩)	K 52	
		66	$S1 \uparrow$		
		67	$S4 \downarrow S5$		
${}_4z_n \times 1 = {}_4z_n$	$S4$	68	⟨49⟩×1	53	
		69	K(1×⟨49⟩)	K 53	
		70	$S4 \uparrow$		
$d_n \times {}_4z_n = \delta y_n$		71	⟨51⟩×⟨53⟩	54	
		72	K(⟨53⟩×⟨51⟩)	K 54	
$d_n y'_n \times (-{}_4z_n) = \delta x_n$		73	⟨52⟩× −⟨53⟩	56	
		74	K(−⟨53⟩×⟨52⟩)	K 56	
x_n	$S6$	D		65	
		75	$S3 \downarrow S6$		
$x_n + \delta x_n = X_n^A$	$S3$	76	⟨55⟩+⟨56⟩	61	
		77	−⟨55⟩ −⟨56⟩	62	
		78	K(⟨61⟩, ⟨62⟩)	K 61	
$x_n - \delta x_n = X_n^d$		79	⟨55⟩ −⟨56⟩	66	
		80	−⟨55⟩ +⟨56⟩	67	
		81	K(⟨66⟩, ⟨67⟩)	K 66	
$y_n + \delta y_n = Y_n^A$		82	⟨50⟩+⟨54⟩	63	
		83	−⟨50⟩ −⟨54⟩	64	
		84	K(⟨63⟩, ⟨64⟩)	K 63	
$y_n - \delta y_n = Y_n^d$		85	⟨50⟩ −⟨54⟩	68	
		86	−⟨50⟩ +⟨54⟩	69	
		87	K(⟨68⟩, ⟨69⟩)	K 68	
			$S5 \uparrow$		

Fig.7.5 - Operation Table for Calculating Fundamental Points of Periphery, Page 2

STAT

Problem: Calculation of Equidistant Coordinates of 33 Profiles
2. Calculation of Fundamental Points of Periphery

Table 2
Page 3

Mathematical Statement	Sel	Op. No.	Operation Symbol: Operation	Operation Symbol: Where	Remarks
	$(S+7)$	88	$S6 \Downarrow S+7$		
X^h_{n+1}		89 90	$\langle 61\rangle \times 1$ $K(1\times\langle 61\rangle)$	75 K 75	
X^h_{n+1}		91 92	$\langle 63\rangle \times 1$ $K(1\times\langle 63\rangle)$	76 K 76	
		93	$S6\uparrow$		
	$(S-7)$	94	$S6 \Downarrow S-7$		
X^d_{n+1}	$S6$	95 96	$\langle 66\rangle \times 1$ $K(1\times\langle 66\rangle)$	75 K 75	
Y^d_{n+1}		97 98	$\langle 68\rangle \times 1$ $K(1\times\langle 68\rangle)$	76 K 76	
		99	$S6\uparrow$		
	$S7$				$S7=(S+7)+$ $+(S-7)$
X_n		100 101	$\langle 75\rangle \times 1$ $K(1\times\langle 75\rangle)$	77' K 77'	T Operation Card to card in Succession
Y_n		102 103	$\langle 76\rangle \times 1$ $K(1\times\langle 76\rangle)$	78' K 78'	T
$X_{n+1}-X_n=\Delta X_n$		104 105 106	$\langle 75\rangle - \langle 77\rangle$ $-\langle 75\rangle + \langle 77\rangle$ $K(\langle 79\rangle, \langle 70\rangle)$	79 70 K 79	
$Y_{n+1}-Y_n=\Delta Y_n$		107 108 109	$\langle 76\rangle - \langle 78\rangle$ $-\langle 76\rangle + \langle 78\rangle$ $K(\langle 71\rangle, \langle 72\rangle)$	71 72 K 71	
$\Delta Y_n : \Delta X_n = Y'_{n,n+1}$		110 111	$\langle 71\rangle : \langle 79\rangle$ $K(\langle 73\rangle \times \langle 79\rangle)$	73 K 73	T

Fig. 7.6 - Operation Table for Calculating Fundamental Points of Periphery, Page 3

STAT

iteration step usually furnishes the accurate value.

The symbol S6 | S + 7 means that set 6 is combined with set (+ 7), i.e., with set 7 for the upper (+) periphery, in such a way that the nth card of set 6 follows the (n - 1)th card of set (+ 7).

The symbol 77' in the column "where" means field No.77 on the next card of the same set.

The symbol T in the column "remarks" means: "print the result with the tabulator"

7.5 Interpolation Method

In the formulation of the interpolation method, the following was taken into consideration: a) The large number of interpolated values (7260 investigated points), b) The time required for carrying out the interpolation, c) The required precision of 0.001 of the thickness of the profile c, d) The requirement that the interpolation curve must have a continuous first derivation, e) The requirement of a "nonundulating" interpolation curve.

With the formulated method, most of the operations are carried out for the preparatory calculation of the interpolation coefficients on 17 cards for each periphery, and hence a total of 66 × 17 = 1122 cards for one set. The actual interpolation on 66 × 110 = 7260 cards contains a minimal number of operations. In this way not only a saving of cards, but especially a considerable saving in time is obtained.

The basis of the method is the interpolation of a parabolic curve. What is here essentially concerned is linear interpolation of a "derivation line". The interpolation parabolic curve is calculated by integration of the "derivation line".

With the 17 points of the periphery (X_n, Y_n) (see Fig.7.7a), we calculate their 16 first proportional differences

$$Y'_{n,n+1} = \frac{Y_{n+1} - Y_n}{X_{n+1} - X_n},$$

which we assign to the abscissa $X_{n-1,n+1} = \frac{1}{2}(X_{n,n+1})$ for $n = 0.1, \ldots, 15$ (see Fig.7.7b). From these 16 first proportional differences we calculated 15 second proportional differences

$$Y''_{n-1,n+1} = \frac{Y'_{n,n+1} - Y'_{n-1,n}}{X_{n,n+1} - X_{n-1,n}}, \quad *$$

which we assign to the abscissas $X_{n-1,n+1} = \frac{1}{2}(X_{n,n+1} + X_{n,\ n+1})$ (see Fig.7.7c).

From these second proportional differences we form the differences from which we select the minimal differences. In Fig.7.7c the minimal difference is $Y''_{3.5} - Y''_{2.4}$. We join these points of minimal difference and we follow a continuous line in both directions in the "second derivation" (Fig.7.7c). This line comprises the abscissas which pass through the points of the second proportional differences $X_{n-1,\ n+1}$, $Y''_{n-1,n+1}$) denoted by small circles.

With the individual abscissas we obtain by integration in the "first derivation" the parabolic curve (dashed curve in Fig.7.7b), which we join together in the points ($X_{n,n+1}$, $Y'_{n,n+1}$), and have in these points a common tangent. However, this parabolic curve, the "1st derivation", is not used for further calculation. Instead, we use the above-mentioned common tangent, which intersects in the points of the abscissas $X_{n-1,n+1}$. This segment of the tangent forms a continuous line. Then by integration we obtain the interpolation parabolic curve joined together in the points (X_n, Y_n) (Fig.7.7a).

In Fig.7.7, for the sake of legibility, the points are denoted by their ordinates. The points belonging to the tangent are therefore denoted by their index.

---

* The precise calculation of the value $Y''_{3.4}$ is not presented, because it is unessential for this case and would make the explanation too complicated

STAT

The points denoted by small circles are given as the result of the preceeding operation (Y_n), or are calculated as the first or the second proportional difference.

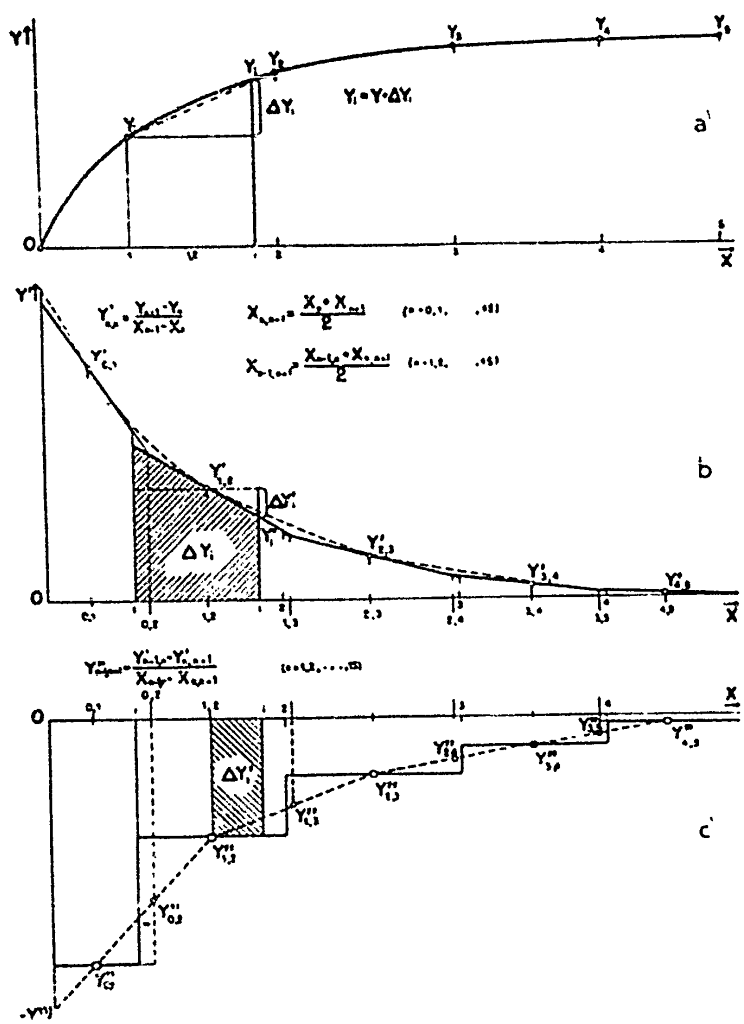


Fig.7.7 - a) Interpolation Curve, b) Its First Derivation, c) Its Second Derivation

$$Y'_{n,n+1} = \frac{Y_{n+1} - Y_n}{X_{n+1} - X_n} \qquad (n = 0, 1, \ldots, 15),$$

$$Y''_{n-1,n+1} = \frac{Y'_{n-1,n} - Y'_{n,n+1}}{X_{n-1,n} - X_{n,n+1}} \qquad (n = 1, 2, \ldots, 15).$$

The points denoted in Fig.7.7c by small squares were calculated by starting with the least difference $Y''_{3.5} - Y''_{2.4}$. The point $Y''_{3.4}$ denoted by solid square is given by the expression

$$Y''_{3,4} = \tfrac{1}{2}(Y''_{2,4} + Y''_{3,5}) \;.^{1)}$$

The points to the left of this point were calculated in succession according to the relations

$$Y''_{2,3} = 2Y''_{2,4} - Y''_{3,4}\,,$$
$$Y''_{1,2} = 2Y''_{1,3} - Y''_{2,3}\,,$$
$$Y''_{0,1} = 2Y''_{0,2} - Y''_{1,2}\,,$$

the points to the right according to the relations

$$Y''_{4,5} = 2Y''_{3,5} - Y''_{3,4}$$
$$Y''_{5,6} = 2Y''_{4,6} - Y''_{4,5}$$

etc

The dashed line in Fig.7.7c is the 1st derivation of the parabolic curve, also denoted by a dashed line in Fig.7.7b. The solid abscissas in Fig.7.7c are the 1st derivations of the tangent to the parabolic curve of Fig.7.7b. This tangent is also a solid line, and is the 1st derivation of the parabolic curve, shown as a solid line in Fig.7.7a.

Figure 7.7c shows that the area $\delta Y_1' = Y''_{1.2}(X_1 - X_{1.2})$ and Fig.7.7b shows that the area $\Delta Y_1 = \frac{1}{2}(Y_1' - Y_1')(X_1 - X_1)$. Also valid are

STAT

/33

$$Y_i' = Y_{1,2}' + \Delta Y_i' = Y_{1,2}' + Y_{1,2}''(X_i - X_{1,2})$$
$$Y_1' = Y_{1,2}' + Y_{1,2}''(X_1 - X_{1,2})$$
$$Y_1' + Y_i' = 2Y_{1,2}' + Y_{1,2}''(X_1 + X_i - 2X_{1,2})$$

and since $X_{1.2} = \frac{1}{2}(X_1 + X_2)$

$$Y_1' + Y_i' = 2Y_{1,2}' - Y_{1,2}''(X_2 - X_i) .$$

The interpolated value Y_i for $X_1 \leq X_i \leq X_2$ is therefore given by the relation

$$Y_i = Y_1 + \Delta Y_i = Y_1 + \tfrac{1}{2}[2Y_{1,2}' - Y_{1,2}''(X_2 - X_i)] . (X_i - X_1) . \quad (7.14)$$

The operations corresponding to pages 1, 2, 3 of Table 3 are presented in Figs.7.8, 7.9, 7.10.

Explanations to Operation Table 3

By transferring the value $Y'_{n,\ n+1}$ from card n to the next card n + 1 (operation 118), we get the value $Y'_{n-1,n}$, i.e., with an index lower by unity. The result of operation 142 is perforated simultaneously into fields 96 and 97 (this value corresponds to the point represented by a solid square in Fig.7.7c. The value in field 96 is used for carrying out the polygonal segment in the 2nd derivation (dashed lines in Fig.7.7c) and the value in field 97 serves for verifying the calculation of the complementary number. The symbol S9m means: "Make up a set consisting only of the cards containing the minimum value of 2ΔY" from each of the subperipheries". The symbol <90> - <96*> means: "from the content of field 90 subtract the content of field 96 of the preceding (*) card of the same set!" The symbol S10 makes reference to the set of main cards (.). This means that, before beginning operation 179, the perforator must perforate into the cards of set 10, which have so far been processed as ordinary cards, the perforation of the main cards. This is the reason for entering on the last line of Table 3, Page 2 the symbol D ... H, which means: "after carrying out operation No.178, perforate the

POOR ORIGINAL

Problem: Calculation of Equidistant Coordinates of 33 Profiles
3. Interpolation

Table 3
Page 1

Mathematical Statement	Set		Op-No.	Operation Symbol: Operation	Operation Symbol: Where	Remarks
	S8		112	S7 ↓ S8		
$Y'_{n,n+1}$		S7	113	⟨73⟩×1	80	
			114	K(1×⟨73⟩)	K 80	
ΔX_n			115	⟨79⟩×1	81	
			116	K(1×⟨79⟩)	K 83	
			117	S7 ↑		
$Y'_{n-1,n}$			118	⟨80⟩×1	82'	operation with card on next card of same set
			119	K(1×⟨80⟩)	K 82'	
ΔX_{n-1}			120	⟨81⟩×1	83'	
			121	K(1×⟨81⟩)	K 83'	
$Y'_{n,n+1} - Y'_{n-1,n} = (Y'_{n,n+1} - Y'_{n-1,n})$			122	⟨80⟩ ⟨82⟩	84	
			123	⟨80⟩ + ⟨82⟩	85	
			124	K(⟨84⟩, ⟨85⟩)	K 84	
$\Delta X_n + \Delta X_{n-1} = 2(X_{n,n+1} - X_{n-1,n})$			125	⟨81⟩ + ⟨83⟩	86	
			126	−⟨81⟩ −⟨83⟩	87	
			127	K(⟨86⟩, ⟨87⟩)	K 86	
$(Y'_{n,n+1} - Y'_{n-1,n}) : 2(X_{n,n+1} - X_{n-1,n}) = 0{,}5 Y''_{n-1,n+1}$			128	⟨84⟩ : ⟨86⟩	88	
			129	K(⟨88⟩×⟨86⟩)	K 88	
	S9		130	S8 ↓ S9		
$0{,}5 \cdot Y''_{n-1,n+1} \times 4 = 2Y''_{n-1,n+1}$		S8	131	⟨88⟩×4	90	
			132	K(4×⟨88⟩)	K 90	
			133	S8 ↑		
$2Y''_{n-2,n}$			134	⟨90⟩×1	91'	operation with card on next card of same set
			135	K(1×⟨90⟩)	K 91'	
$2Y''_{n-1,n+1} - 2Y''_{n-2,n} = 2\Delta Y''$			136	⟨90⟩−⟨91⟩	92	Pick out $2\Delta Y''$ - min.
			137	−⟨90⟩ + ⟨91⟩	93	
			138	K(⟨92⟩, ⟨93⟩)	K 92	
$2Y''_{n-1,n+1} + 2Y''_{n-2,n} = 4Y''_{n-1,n}$	S9m		139	⟨90⟩ + ⟨91⟩	94	Operation only on cards containing $2\Delta Y''_{min.}$ *)
			140	−⟨90⟩ − ⟨91⟩	95	
			141	K(⟨94⟩, ⟨95⟩)	K 94	
$4Y''_{n-1,n} \times 0{,}25 = Y''_{n-1,n}$			142	⟨94⟩×0,25	96,97	
			143	K(0,25×⟨94⟩)	K 96,97	

* Set 9 is divided into two partial sets! Partial set $\overline{9}$ containing the cards with indices 0, 1, 2 ... to indices with less than the cards containing the value $Y''_{min.}$ are arranged in descending order. Partial set 9 containing the remaining cards with the cards containing the value $2\Delta Y''_{min}$ are left in ascending order. Operations 139-143 are carried out only on the cards containing $2\Delta Y''_{min}$ (S9m).

Fig.7-8 - Operation Table for Interpolation, Page 1

STAT

Problem: Calculation of Equidistant Coordinates of 33 Profiles
3. Interpolation

Table 3
Page 2

Mathematical Statement	Set	Op. No.	Operation Symbol: Operation	Operation Symbol: where	Remarks
	$\overline{S9}$	144	$S9m \downarrow \overline{S9}$		
$\overline{2Y''_{n-1,n+1} \quad Y''_{n,n+1} \quad Y''_{n\ n,1}}$	S9m	145	⟨90⟩ — ⟨96*⟩	96	Operation along pack of cards
		146	—⟨90⟩ + ⟨97*⟩	97	
		147	K(⟨96⟩, ⟨97⟩)	K 96	
		148	S9m ↑		
	S9	149	S9m ↓ S9		
$2Y''_{n\ 2,n} \quad Y''_{n-1,n\ 1} \quad Y''_{n\ 1,n}$	S9m	150	⟨91⟩ 96*⟩	96	Operation along pack of cards
		151	⟨91⟩ + ⟨97*	97	
		152	K(⟨96⟩, ⟨97⟩)	K 96	
	S9	153	Rearrange cards according to n		
	S10	154	S9 ⇓ S10		
$\overline{Y''_{n,n+1}}$	S9	155	1×⟨96⟩	100	
		156	K(⟨96⟩×1)	K100	
		157	S9 ↑		
		158	S8 ↓ S10		
$0{,}5\Delta X_n$	S8	159	0,5×⟨81⟩	101	
		160	K(⟨81⟩×0,5)	K101	
$\overline{Y'_{n,n+1}}$		161	1×⟨80⟩	102	
		162	K(⟨80⟩×1)	K102	
		163	S8 ↑		
$\overline{Y''_{n,n+1}} \times \overline{0{,}5\Delta X_n} \quad 0{,}5Y''_{n,n+1} \cdot \Delta X_n$		164	⟨100⟩×⟨101⟩	103	
		165	K(⟨101⟨×⟩100⟩	K103	
$\overline{Y'_{n,n+1}} \quad \overline{0{,}5Y''_{n,n+1} \cdot \Delta X_n} \quad \overline{Y'_n}$		166	⟨102⟩—⟨103⟩	104	
		167	—⟨102⟩+⟨103⟩	106	
		168	K(⟨104⟩, ⟨106⟩)	K104	
		169	S7 ↓ S10		
$\overline{X_n}$	S7	170	1×⟨77⟩	105	
		171	K(⟨77⟩×1)	K105	
$\overline{X_{n+1}}$		172	1×⟨75⟩	107	
		173	K(⟨75⟩×1)	K107	
$\overline{Y_n}$		174	1×⟨78⟩	108	
		175	K(⟨78⟩×1)	K108	
		176	S7 ↑		
$\overline{X_n}$		177	1×⟨105⟩	109	
		178	K(⟨105⟩×1)	K109	
		D		H	

Fig.7.9 – Operation Table for Interpolation, Page 2

STAT

Problem: Calculation of Equidistant Coordinates of 33 Profiles
3. Interpolation

Table 3
Page 3

Mathematical Statement	Set	Op.No.	Operation Symbol: Operation	Operation Symbol: Where	Remarks
X_i	S11	D			X_i = 0,000; 0,005; 0,01; 0,015, ...; 0,095; 0,10; 0,11; ... ...; 0,99; 1,00.
		179	$\dot{S}10 \downarrow S11$		
$X_{n+1}-X_i$	$\dot{S}10$	180	⟨107⟩ ⟨115	111	
		181	⟨107⟩ ⟨115	110	
		182	K(⟨111⟩, ⟨110⟩)	K111	
$Y''_{n,n+1}\times(X_{n+1}-X_i)$		183	⟨100⟩×⟨111⟩	112	
		184	K(⟨111⟩×⟨100⟩)	K112	
$2 Y'_{n,n+1}$		185	2×⟨101	113	
		186	K(⟨102⟩×2)	K113	
Y'_n		187	1×⟨104⟩	114	
		188	K(⟨104⟩×1)	K114	
		189	$\dot{S}10\uparrow$		
$2Y'_{n,n+1}-Y''_{n,n+1}\cdot(X_{n+1}-X_i)$		190	⟨113⟩ ⟨112	116	
		191	⟨113⟩+⟨112	117	
		192	K(⟨116⟩, ⟨117⟩)	K116	
$[2Y'_{n,n+1}-Y''_{n,n+1}\cdot(X_{n+1}-X_i)]$ $Y'_n=Y'_i$		193	⟨116⟩ ⟨114	118	
		194	⟨116⟩+⟨114	119	
		195	K(⟨118⟩, ⟨119⟩)	K118	
X_i	S12	D		125	
		196	$S11 \downarrow S12$		
$[2Y'_{n,n+1}-Y''_{n,n+1}\cdot(X_{n+1}-X_i)]:2$	S11	197	0,5×⟨116⟩	120	
		198	K(⟨116⟩×0,5)	K120	
		199	$S11\uparrow$		
		200	$\dot{S}10 \downarrow S12$		
X_i-X_n	$\dot{S}10$	201	⟨125⟩ ⟨109⟩	121	
		202	⟨125⟩ ⟨109⟩	122	
		203	K(⟨121⟩, ⟨122⟩)	K121	
Y_n		204	1×⟨103⟩	123	
		205	K(⟨104⟩×1)	K123	
		206	$\dot{S}10\uparrow$		
$[Y'_{n,n+1}-\frac{1}{2}Y''_{n,n+1}(X_{n+1}-X_i)]\times(X_i-X_n)$		207	⟨120⟩×⟨121⟩	124	
		208	K(⟨121⟩×⟨120⟩)	K124	
$[Y'_{n,n+1}-\frac{1}{2}Y''_{n,n+1}(X_{n+1}-X_i)]\cdot(X_i-X_n)+Y_n=Y_i$		209	⟨124⟩+⟨123⟩	126	
		210	⟨124⟩ ⟨123⟩	127	
		211	K(⟨126⟩, ⟨127⟩)	K126	

Fig.7.10 - Operation Table for Interpolation, Page 3

[redacted] STAT

perforation of the main cards into the cards of set 10!"

7.6 Final Calculation

The calculated interpolation point (X_i, Y_j) plotted against the value c_N/c^* is perforated into N main cards. The next step is the reduction of the peripheral coordinates of the profile having the depth $c^* = 1$ to the required depth c_N. The plotting of the equidistant curve in the distance of the radius r of the working tool is carried out in the same way as the plotting of the profile on the center line. The transformation of the equidistant coordinates to the given coordinate system occurs with the help of the main cards, into which have already been perforated the values X_o^N, Y_o^N sin γ^N, cos γ^N. (N is the number of profiles; X_o^N, Y_o^N are the coordinates of the entrance corner of the profile in the selected coordinate system, and γ^N is the angle between the joining line of the entrance corner and the exit corner of the profile and the X axis of the selected coordinate system).

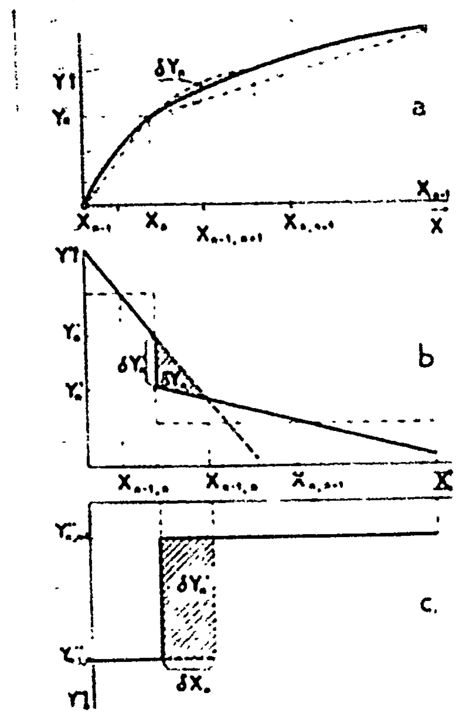


Fig.7.11 - Interpolation Line and its 1st and 2nd Derivations about the Point (X_n, Y_n)

7.7 Error of the Employed Interpolation Method

The errors occurring with this interpolation method are of two kinds. The one kind is due to the fact that the abscissas X_n are not equidistant, which is due to the fact that $X_{n-1,\ n+1} \neq X_n$. Figure 7.11 shows a segment of the interpolation

271

line about the point (X_n, Y_n) and its first and second derivations. The two interpolation curves do not have a common tangent in the point (X_n, Y_n). The directions of the two tangents differ by $\delta Y'_n$. The tangent of the prolonged left parabolic curve in the point of the abscissa $X_{n-1,n+1}$ is parallel with the tangent of the right curve in the same abscissa. The absolute value of the difference between the ordinates of the two points is denoted δY_n, and we regard this error as due to the fact that $X_n \neq X_{n-1,b+1}$.

The Figure shows that

$$\delta Y'_n = |\delta X_n| \cdot |Y''_{n,n+1} - Y''_{n-1,n}| = |\delta X_n| \cdot |\Delta Y''_{n-1,n}|$$

and that

$$\delta Y_n = \tfrac{1}{2}|\delta Y'_n| \cdot |\delta X_n| = \tfrac{1}{2}(\delta X_n)^2 \cdot |\Delta Y''_{n-1,n}| .$$

With regard to the indexing of the coordinates and the difference

$$\delta X_n = X_{n-1,n+1} - X_n = \frac{\Delta X_n - \Delta X_{n-1}}{4}$$

we get the same values as listed in the Table of Fig.7.12.

Figure 7.12 gives the fundamental values of the center line and the upper periphery of the profile N = 1. The columns 1 and 10 contain the indices of the fundamental points, columns 2 and 4 the coordinates of the center line, and columns 5 and 7 the coordinates of the upper periphery. Column 8 contains the first proportional differences, column 9 the two-fold second proportional differences, and column 11 the calculated "second derivation" of the interpolation curve in the interval (X_n, X_{n+1}). Column 12 contains the differences ($\Delta Y''_{n-1,n} = Y''_{n,n+1} - Y''_{n-1,n}$), column 13 the second differences ($\Delta^2 X_{n-1} = \Delta X_n - \Delta X_{n-1}$), column 14 the values $(\delta X_n)^2 \cdot 10^8 = \frac{1}{4}(\Delta^2 X_{n-1})^2 \cdot 10^8$ and column 15 the corresponding errors δY_n.

The greatest errors occur where the intervals between abscissas of the center

line are doubled, i.e., in points 2, 5, and 7, or halved, i.e., in point 14. Also in point 1 occurs a considerable error due to the fact that $\Delta Y'_{0.1}$ is large. However, the error does not reach even 1/10 of the permissible. (The maximal permissible error in this problem is 1-3 decimals).

1	2	3	4	5	6	7	8	9
	a)			b)				
n	x_n	Δx_n	y_n	X_n	ΔX_n	Y_n	$Y'_{n,n+1}$	$2Y''_{n-1,n+1}$
0	0,000 000	0,012 5	0,000 000	0,000 000	0,008 131	0,000 000	2,205 52	282,015 96
1	0,012 500	0,012 5	0,003 785	0,008 131	0,010 778	0,018 421	0,932 30	31,255 52
2	0,025 000	0,025	0,007 404	0,018 909	0,023 072	0,028 470	0,007 80	10,492 12
3	0,050 000	0,025	0,014 502	0,041 981	0,023 800	0,043 879	0,544 74	5,956 32
4	0,075 000	0,025	0,021 113	0,065 847	0,024 328	0,056 880	0,472 96	1 673 00
5	0,100 000	0,050	0,027 208	0,090 185	0,019 738	0,068 391	0,380 42	3,237 20
6	0,150 000	0,050	0,038 388	0,139 923	0,050 644	0,087 611	0,305 18	2,643 36
7	0,200 000	0,1	0,047 772	0,190 567	0,103 023	0,103 067	0,203 63	2,233 80
8	0,300 000	0,1	0,061 422	0,293 590	0,104 194	0,124 040	0,087 93	1,953 48
9	0,400 000	0,1	0,068 246	0,397 784	0,103 600	0,133 209	0,013 58	1,567 76
10	0,500 000	0,1	0,068 529	0,501 444	0,102 541	0,131 801	0,094 40	1,579 72
11	0,600 000	0,1	0,063 961	0,603 985	0,101 608	0,122 121	0,175 05	1,550 24
12	0,700 000	0,1	0,054 824	0,705 653	0,100 386	0,104 323	0,253 36	1,548 72
13	0,800 000	0,1	0,041 118	0,806 039	0,098 772	0,078 889	0,330 47	1,703 20
14	0,900 000	0,050	0,022 841	0,904 811	0,048 644	0,040 247	0,393 21	1,621 52
15	0,950 000	0,050	0,011 993	0,953 455	0,018 128	0,027 118	0,432 47	
16	1,000 000		0,000 001	1,001 583		0,006 301		

10	11	12	13	14	15	16	17	18
n	$Y''_{n,n+1}$	$\Delta Y''_{n-1,n}$	$\Delta^2 X_{n-1}$	$(\delta X_n)^2 \cdot 10^6$	δY_n	$(\Delta X_n)^2/4$	[illegible]	[illegible]
0	258,332				0,000 052			
1	23,084	234,65	0,002 647	44	0,000 076			
2	7,571	16,11	0,012 294	947	0,000 000	0,000 112	1,65	0,000 33
3	2,021	4,65	0,000 794	4	0,000 000	0,000 147	1,40	0,000 10
4	3,035	0,11	0,000 462	1	0,000 000	0,000 018	1,40	0,000 43
5	1,038	1,40	0,025 410	4 030	0,000 000	0,000 041	0,56	0,000 18
6	1,600	0,04	0,000 906	5	0,000 047	0,002 653	0,56	0,000 74
7	1,044	0,55	0,052 379	17 148	0,000 000	0,002 714	0,43	0,000 58
8	1,190	0,15	0,001 171	9	0,000 000	0,002 086	0,43	0,000 58
9	0,764	0,43	0,000 534	2	0,000 000	0,002 629	0,04	0,000 05
10	0,804	-0,04	0,001 119	8	0,000 000	0,002 584	0,03	0,000 04
11	0,776	0,03	0,000 873	5	0,000 000	0,002 519	0,00	0,000 00
12	0,775	0,00	-0,001 282	10	0,000 000	0,002 439	0,16	0,000 20
13	0,774	0,00	-0,001 014	16	0,000 011	0,000 502	0,24	0,000 07
14	0,020	-0,16	0,050 128	15 705	0,000 000	0,000 579	0,24	0,000 07
15	0,692	0,24	0,060 316	2				
16								

Fig.7.12 - Values for Center Line and Upper Periphery of the Profile N=1. Values for Estimating the Error

a) Center line; b) Upper periphery

91

The second source of error in the circumstance is that Y'' oscillates moderately (see, for example, Fig.7.12, columns 11 and 12). In this case the absolute value of the error may be determined by estimation. The formula for the interpolation of the ordinate Y_i according to eq.(7.14) is

$$Y_i = Y_n + \tfrac{1}{2}[2Y'_{n,n+1} - Y''_{n,n+1}(X_{n+1} - X_i)] \,.\, (X_i - X_n)\,, \qquad (7.15)$$

where

$$X_n \leq X_i \leq X_{n+1}\,.$$

The ordinates of the points Y_n and the directions of the joining lines between adjacent points $Y_{n,n+1}$ are given in advance ($n = 0, 1, 2 \ldots, 16$). The absolute error due to the oscillation of Y'' is denoted by $\delta^* Y_i$. If $Y_{n,n+1}$ differs from the correct value by $\delta Y''_{n,n+1}$, then it follows from eq.(7.15) that

$$\delta^* Y_i = \tfrac{1}{2}|\delta Y''_{n,n+1}(X_{n+1} - X_i) \,.\, (X_i - X_n)|\,.$$

Putting $X_{n+1} = X_n + \Delta X_n$ and $X_i - X_n = m\,\Delta X_n$ for $0 \leq m \leq 1$ gives

$$\delta^* Y_i = \tfrac{1}{2}|\delta Y''_{n,n+1}| \,.\, (1 - m) \,.\, m \,.\, (\Delta X_n)^2\,.$$

In the interval (X_n, X_{n+1}) it is obvious that $\delta^* Y_i$ is the maximum for $m = \frac{1}{2}$. Therefore

$$\delta^* Y_{max} = \tfrac{1}{2}|\delta Y''_{n,n+1}| \,.\, (\Delta X_n)^2/4\,. \qquad (7.16)$$

Assuming that the correct value of Y'' lies somewhere between the calculated values of $Y''_{n,n+1}$ and the adjacent value, e.g., $Y''_{n,n-1,n}$ or $Y''_{n+1,n+2}$. Thus we have $\delta Y''_{n,n+1} < \Delta Y''_{n-n,n}$ or $\delta Y''_{n,n+1} < \Delta Y''_{n,n-1}$. From this it follows that

$$\delta^* Y_{max} \leq \tfrac{1}{8}(\Delta X_n)^2 \operatorname{Max}\,(|\Delta Y''_{n-1,n}|;\ |\Delta Y''_{n,n+1}|) = A\,.$$

Tabulated in Fig.7.12 are: in column 16 for individual points the corresponding values of $(\Delta X_n)^2/4$, in column 17 the larger of the two values $|\Delta Y''_{n-1,n}|$

and $|\Delta Y''_{n,n+1}|$ and in column 18 the values of Λ on the right side of the inequalities, which are smaller than the maximal permissible error 0.001c (c = 1). The main cause of the oscillating of the second derivation of the profile periphery is the circumstance that the symmetrical profile (which is plotted on the center line), on the basis of the tabulated data, already has oscillation of the second derivation of the second periphery in the interval of the required precision.

The entire calculation is carried out on the full number of places (7 digits), because the operational speed of the machine is independent of the number of places of the processed numbers, and to prevent any influence of the error due to the accumulation of rounded errors on the result for the required number of places.

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

LABORATORY OF MATHEMATICAL MACHINES, CZECHOSLOVAK ACADEMY OF SCIENCES

SYMPOSIUM I.

Mathematical Machines

The first part of the Symposium contains results of the research in numerical calculation methods suitable for solution of problems on the Czechoslovak automatic computer SAPO. Coding and a symbolism suitable to formulate instructions for the machine are explained.

In this part of the Symposium a method is described how to form detailed instructions (the programming) for the machine in accordance with a given problem. The use of the method is illustrated by several examples.

Chapter I. describes the general character of a modern automatic computer. Classical methods of numerical calculus are compared with methods suitable for an automatic computer. The possibility of mathematical experiments is pointed out.

Basic concepts of automatic computing, including terms as instruction, address, operation, word, flow sign, are described, and the concept of the instruction net as well. The suitability of symbolism introduced to simplify the development of instruction nets is illustrated by a simple example.

A simplified diagram of the Czechoslovak automatic computer is included, with a description of its principal parts.

Chapter 2. In the first part of this Chapter the coding of data for the Czechoslovak automatic computer is described first from a general point of view.

A detailed study of the coding follows, demonstrating the possibility to express numbers given either in binary or in decimal form. The numerical capacity of the machine is indicated.

The part which follows deals with instruction coding, i.e., with the translation of orders into proper symbols. A complete list of the operational code is

[redacted]

STAT

264

included. Individual operation symbols, their importance and mutual relations are discussed. Operations of a more complicated character are illustrated by examples.

A list of basic operations which is attached should serve as a guide for the preparation of instruction nets.

Chapter 3. introduces a procedure in designing an instruction net: selecting a numerical method, designing the instruction net in a general form, finishing the instruction net in detail. It shows how to fill up blanks used to design the instruction net in its general form and blanks for its detailed form. The Chapter is concluded by a simple example illustrating the procedure described.

Chapter 4. presents an example of designing the instruction net for ray tracing through a centered optical system. The ray tracing represents the most important and laborious part of computations required to reduce aberrations of an optical system by means of variations of its parameters. The usual practice up to now confined the computation to paraxial rays as the tracing of skew rays was too laborious even with a table calculator. An instruction net for the computation of 168 rays has been designed, most of the rays being skew to the optical axis.

First the geometrical analysis of the problem is carried out and the way is described how to choose the starting points of rays and directions of these at the point of entry in the optical system. Remarks about some restraining physical conditions are included together with explanations how to modify the usual numerical calculating procedure to make it suitable for the automatic computer. The Chapter concludes with the detailed description of the instruction net and with explanation of the role of individual instruction blocks. The net is illustrated by a flow diagram.

Chapter 5 is a demonstration of solving of differential equations with the automatic computer. The purpose of the Chapter is to show special devices which may be used when planning instruction nets for an automatic computing machine.

As an illustration a very simple example has been chosen in order that the

STAT

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

essential principles would not be hidden by the complexity of the problem. To solve it, however, a more complicated method is used on purpose, one that can be applied even to a very complex systems of differential equations.

The solution of the equation

$$y'' = P_3(x) + F(t)$$

is worked out by the Runge-Kutta method ($P_3(x)$ is a polynom of third degree and $F(t)$ is a rational function), the initial condition being $x = x_0$, $\frac{dx}{dt} = v_0$, for $t = t_0$. Such a method has been chosen because it is homogeneous (an example of a nonhomogeneous method is for instance, the Adams method, which requires quite a different kind of computation at the beginning than at the later stages of the process) and because the number of intermediate results is comparatively small; both these facts allow to get the solution of comparatively complex problems with a low storage requirement. Thanks to these circumstances the method can be considered as well-suited for an automatic computer.

Instruction nets for quite complex systems of differential equations can be prepared by a generalization of the procedure. The time needed to accomplish 50 Runge-Kutta steps is estimated for the problem under consideration.

The second part of the Symposium deals with the use of Czechoslovak punched card machines for numerical solutions of mathematical problems.

Chapter 6. The first paragraphs describe punched cards with the symbolism used to express instructions for the working procedure and explain manipulations with punched cards and operations with numbers that occur most frequently. In the remaining paragraphs some punched card machines are briefly described: the punch, the sorting machine, the tabulator, and the calculating punch which adds, subtracts, multiplies and divides with respect to signs.

Chapter 7. A practical engineering problem has been solved with punched card machinery. Coordinates of points outlining 33 cross sections of turbocompressor

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

blades have been computed as well as coordinates of points belonging to curves equidistant to those cross sections to facilitate the manufacture of blades on a milling machine.

In this Chapter only a part of the whole problem is presented especially a suitable interpolation method giving the required accuracy. Operation tables are given expressing instructions for the procedure.

In the last paragraph some estimates of errors in computation are discussed.

STAT

75

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

TABLE OF CONTENTS

STAT

Page

[redacted]

STAT

187

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

Page

STAT

167

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

Page

61

Declassified in Part - Sanitized Copy Approved for Release 2012/11/20 : CIA-RDP81-01043R001400020002-8

Operational Mark	Operational Symbol
S	$\langle i\rangle + \langle j\rangle \rightarrow \langle k\rangle$
N	$\langle i\rangle \cdot \langle j\rangle \rightarrow \langle k\rangle$
D	$\langle i\rangle : \langle j\rangle \rightarrow \langle k\rangle$
T	$T1\langle i\rangle \rightarrow \langle k\rangle$ $T2\langle j\rangle \rightarrow \langle k\rangle$
M	$-\langle i\rangle$
I	$\lvert\langle i\rangle\rvert$
J	$\lvert\langle j\rangle\rvert$
K	$\lvert\ldots\ldots\rvert \rightarrow \langle k\rangle$
G	$G \ldots.$
H	$H1 \ldots$ $H2 \ldots$
ST	$\langle i\rangle \underline{+} \langle j\rangle \rightarrow \langle k\rangle$
NT	$\langle i\rangle \underline{\cdot} \langle j\rangle \rightarrow \langle k\rangle$
DT	$\langle i\rangle \underline{:} \langle j\rangle \rightarrow \langle k\rangle$
SX	$\mathrm{Exp}\,\langle j\rangle + \langle i\rangle \rightarrow \langle k\rangle$
WX	$\langle i\rangle = \mathrm{Exp}\,\langle j\rangle \rightarrow \langle k\rangle$
SYZ	$\mathrm{Sgn}\,\langle j\rangle \rightarrow \langle k\rangle$
WYZ	$\langle i\rangle = \mathrm{Sgn}\,\langle j\rangle \rightarrow \langle k\rangle$
SXY	$A!\langle j\rangle + \langle i\rangle \rightarrow \langle k\rangle$
SZ	$B!\langle j\rangle + \langle i\rangle \rightarrow \langle k\rangle$
SY	$C!\langle j\rangle + \langle i\rangle \rightarrow \langle k\rangle$
$SWXY$	$A!\langle j\rangle + \langle i\rangle \rightarrow !\langle k\rangle$
SWZ	$B!\langle j\rangle + \langle i\rangle \rightarrow !\langle k\rangle$
SWY	$C!\langle j\rangle + \langle i\rangle \rightarrow !\langle k\rangle$
WXY	$\langle i\rangle = A!\langle j\rangle \rightarrow !\langle k\rangle$
WZ	$\langle i\rangle = B!\langle j\rangle \rightarrow !\langle k\rangle$
WY	$\langle i\rangle = C!\langle j\rangle \rightarrow !\langle k\rangle$
(Empty)	STOP

2.9 - General Survey of Fundamental Operations

STAT

10